UNIVERSITÀ
DI PAVIA

Università degli Studi di Pavia
Università della Svizzera Italiana

Joint PhD program in Computational Mathematics and Decision Sciences
XXXV cycle

Efficient Models and Algorithms for Image Processing for Industrial Applications

Advisors:
Prof. Stefano Gualandi
Dr. Gabriele Lombardi

PhD Dissertation of:
Andrea Codegoni

Academic year 2021-2022

Abstract

Image processing and computer vision are now part of our daily life and allow artificial intelligence systems to see and perceive the world similarly to humans. Behind the remarkable results obtained by modern computer vision algorithms, there are algorithms whose complexity, in some cases, requires the use of dedicated hardware. However, especially in the industrial field for embedded applications, it is not always possible to use hardware with sufficient computing capacity to manage algorithms of high computational complexity with execution times compatible with industrial needs. In this thesis, we develop computer vision algorithms and methods with low computational complexity and high performances. In the first approach presented, we study the relationship between Fourier-based metrics and Wasserstein distances to propose alternative metrics to the latter, considerably reducing the time required to obtain comparable results. For the second case, instead, we start from an industrial problem and develop a deep learning model for change detection called TinyCD, obtaining state-of-the-art performance and reducing the computational complexity required by at least a third compared to the existing literature.

Research activities

Publication on refereed international journals

- Auricchio, G., Codegoni, A., Gualandi, S., Toscani, G., Veneroni, M. (2020). The equivalence of Fourier-based and Wasserstein metrics on imaging problems. Rendiconti Lincei, 31(3), 627-649.

- Auricchio, G., Codegoni, A., Gualandi, S., Zambon, L. (2021). The Fourier Discrepancy Function. Communications in Mathematical Sciences. Accepted on July 2022. `https://intlpress.com/site/pub/pages/journals/items/cms/_home/acceptedpapers/index.php`

- Codegoni, A., Lombardi, G., Ferrari, A. (2022). TinyCD: A (Not So) Deep Learning Model For Change Detection. Neural Computing and Applications. Published Online December, 18 2022. `https://link.springer.com/article/10.1007/s00521-022-08122-3` Reproduced with permission from Springer Nature. Code and datasets are available here: `https://github.com/AndreaCodegoni/Tiny_model_4_CD`

Publication on refereed international conferences

- Codegoni, A., Gualandi, S., Ricciato, F. (2022) On the Application of the Fourier-based Distance to Spatial Statistics. Conference on New Techniques and Technologies for Statistics, Eurostat. 7-9 March 2023, Bruxelles.

Pre-print

- Bellazzi, R., Codegoni, A., Gualandi, S., Nicora, G., Vercesi, E. (2021). The Gene Mover's Distance: Single-cell similarity via Optimal Transport. arXiv preprint arXiv:2102.01218.

Teaching activities

- **Teaching assistant:** Numerical Methods - Nonlinear Optimization, 3 ECTS, Bioengineering, University of Pavia, Fall semester 2019-2020, 2020-2021

- **Student Supervision:** Aspetti algebrici e geometrici della programmazione lineare e lineare intera. Mattia Repossi - Bachelor degree in Mathematics. Co-rapporteur. April 2021

Editorial activities

- **Reviewer:** INFORMS Journal Of Computing - from April 2021
- **Organizer:** Conference of Young Applied Mathematicians, 13/17 September 2021, Santa Maria di Leuca (LE), Italia; 18/22 September 2022, Arenzano (GE), Italia

Contents

Chapter 1

Introduction

Image processing plays a fundamental role in our daily life. Our brain can process images effortlessly, and these processed images represent a fundamental piece of our perception of the surrounding environment. In a world in which artificial intelligence is gaining ground, image processing becomes a fundamental element for perceiving the world for artificial intelligence systems. Not surprisingly, image processing and computer vision have been a fervent field of research since the 1960s when the advent of the first computers with sufficient computing power allowed researchers at the American Jet Propulsion Laboratory to improve the image quality of the lunar soil and thanks to other geometric correction and registration techniques, reconstruct the entire lunar surface [1]. Since that time numerous techniques have been developed not only to process and improve image quality but also to extract as much information as possible to make artificial intelligence algorithms able to perceive the world through artificial vision. Self-driving cars and drones can recognize road signs and possible dangers to safely navigate the surrounding world [2–6], you can unlock your phone using a front camera and face recognition algorithms [7, 8]; computer vision is used as a tool to help doctors making the diagnosis [9–11], and in the manufacturing industry, automatic defect detection is employed [12–14]. And these are just a few of the numerous applications of computer vision which have an impact on our everyday life [15]. All this has been made possible also thanks to the technology that has developed dedicated hardware such as GPUs, NPUs, FPGAs and TPUs.

However, the state-of-the-art models and algorithms in computer vision and image processing have heavy computational costs [16]. High computational costs and dedicated hardware can be a problem when we want to apply the research results in industrial applications, which often imposes constraints in terms of hardware and available computational resources.

The common thread of the research carried out in this thesis is the development of algorithms with low computational cost but keeping the performances unchanged, or whose results are comparable to those of the state-of-the-art methods.

In Chapter 2, we follow a mathematical approach leveraging the concept of equivalence between metrics on probability space. From a mathematical point of view, typical problems from image processing and computer vision are always described as a mathematical problem and then analyzed: the reconstruction of images corrupted by noise can be formalized as a minimization problem and, thanks to the calculus of variations, algorithms to find the minimum, i.e.

the denoised image, are constructed [17–19]; segmentation problems can be viewed as stochastic problems or graph problem and tackled with stochastic method [20–22] or linear programming tools [23–25]; moving objects can be modelled with partial differential equations and gradient flows [26–29].

It is fascinating to note how often the mathematical tools that are used to tackle computer vision problems were not originally conceived for that purpose, but thanks to the flexibility of mathematics they have been adopted. For example, the Fourier transform was theorized by Jean Baptiste Joseph Fourier in his treatise *Théorie analytique de la chaleur* [30]. Then the Fourier transform, the Fourier series and the whole Fourier analysis became a very important field of research in a lot of mathematical fields such as partial differential equations, probability theory, complex analysis, and differential geometry [31]. Also, Fourier analysis plays a crucial role in engineering since it is widely applied in signal processing and image processing [32, 33].

Similarly, Optimal Transport, born from the intuitions of Monge and formalized for the first time in his work *Mémoire sur la théorie des déblais et des remblais* of 1781 [34], became a topic of great interest in various fields of mathematics such as probability, partial differential equations, kinetic theory and differential geometry [35, 36]. More recently, optimal transport has also been studied as a tool for image processing, finding applications in image retrieval, colour transfer, image recognition, and image generation [37].

Fourier analysis and optimal transport are related from the point of view of measure theory. In fact, through the Fourier transform it is possible to define a family of distances which are topologically equivalent to the family of distances induced by the optimal transport, the Wasserstein distances [35]. This link turns out to be very useful for example in the field of kinetic theory where weak convergence is used to study the asymptotic behaviour of the solutions of kinetic equations such as Boltzmann's equations [38–40]. Weak convergence is usually studied through the Wasserstein distances but, in this case, the Fourier distances turn out to be a more appropriate and manageable tool and, thanks to the equivalence, the conclusions are the same. Despite these connections, the relationship between Wasserstein distance and Fourier metrics has received little attention in image processing.

For this reason, in Chapter 2 we study the link between these two distances in a suitable setting for image processing. After a brief introduction to Optimal Transport and Wasserstein distances conducted in Section 2.1, we introduce the Fourier-based metrics in Section 2.2 and we extend in Section 2.3 the equivalence result between Fourier-based metrics and Wasserstein distance of order two in the general setting. Then, in Section 2.4 we analyze the Fourier-based metrics in the discrete setting, and we derive for these cases explicit equivalence constants with the Wasserstein distance. To cope with applications, in Section 2.5 we review the properties of the discrete Fourier Transform and then, in Section 2.6 we present our numerical results. In particular, Section 2.6.1 contains a benchmark between Fourier-based metrics and Wasserstein distances showing the computational advantages of Fourier-based metrics and highlighting that the relationships between these two families of metrics are, in practice, stricter than the ones expressed by the formal equivalence results. To validate

these claims, in Section 2.6.2, we do the same comparison in a realistic dataset showing that the Fourier-based metrics could be successfully used in applications where the Wasserstein distance is used to evaluate the goodness of fit between reconstructed probability measures and ground truth probability measures. Finally, in Section 2.6.3, we apply the Fourier-based metrics also in an anomaly detection task to highlight its performances also on Optimal Transport non-related applications.

Chapter 3, on the other hand, takes its cue from an industrial problem and was a great opportunity to deepen and study image processing and computer vision with Deep Learning tools. The history of deep learning in image processing can be traced back to the late 1950s when neurophysiologists discovered through experiments on animals that sight operates by layering knowledge [41]. Thus, layered models began to come to life. However, we have to wait until the late 90s, and early 2000s to see deep learning models in computer vision take root [42]. Unlike mathematical methods for image processing, deep learning models learn directly from data and also need large computational capabilities or dedicated hardware such as GPUs. When computers became powerful enough to support these models, and appropriate datasets were introduced and standardized, deep learning moved the performance even beyond human capabilities, and today represents the state of the art in image processing and computer vision [43–45].

Even if computational resources and datasets are no longer an obstacle in research today, they still represent a problem in the industrial field [16]. Collecting datasets for specific applications turns out to be a very time-consuming task. Furthermore, dedicated hardware can cost several thousand euros, while to keep production costs low, we would like to use low-cost devices with limited computational resources, without sacrificing performance. To better understand the industrial scenario that we are facing, we open Chapter 3 describing in detail the Line Clearance, namely the problem of monitoring the production line state and detecting possible hazards, pointing out limitations and particular requirements imposed by the industrial setting. To overcome the difficulties related to the creation of the dataset and develop a model, we decided to tackle a problem similar to the industrial one, but coming from the world of change detection for aerial images [46, 47]. This workaround is also useful to validate our model out of the particular industrial application, giving our model general applicability. We introduce the change detection on aerial images in Section 3.1, and we review the existing literature in Section 3.2. Section 3.4 contains the description of TinyCD, our proposed model. The philosophy of our approach is to develop a model that uses low-level local features to compare two different images and track the unwanted changes. As reported in Section 3.5, the proposed model has reduced the computational complexity to one-third and has at least one-twelfth the number of parameters compared to the current state-of-the-art models for aerial change detection. We discuss the results in Section 3.6, and then we conclude the chapter with Section 3.7 showing how TinyCD performs in the industrial scenario.

Finally, we conclude our work in Chapter 4, giving perspectives on future works.

Parts of the results presented in this thesis are contained in papers that have been submitted by the author and coauthors in peer reviewed journals or peer reviewed conferences:

Chapter 2

- Auricchio, G., Codegoni, A., Gualandi, S., Toscani, G., Veneroni, M. (2020). The equivalence of Fourier-based and Wasserstein metrics on imaging problems. Rendiconti Lincei, 31(3), 627-649.
- Auricchio, G., Codegoni, A., Gualandi, S., Zambon, L. (2021). The Fourier Discrepancy Function. Communications in Mathematical Sciences. Accepted on July 2022. `https://intlpress.com/site/pub/pages/journals/items/cms/_home/acceptedpapers/index.php`
- Codegoni, A., Gualandi, S., Ricciato, F. (2022). On the Application of the Fourier-based Distance to Spatial Statistics. Conference on New Techniques and Technologies for Statistics, Eurostat. 7-9 March 2023, Bruxelles.

Chapter 3

- Carrioli, L., Codegoni, A., Lombardi, G. (2022). One-shot anomaly segmentation for Line Clearance. CompMat 2022, Spring Workshop. March, 16-17 2022, Pavia.
- Codegoni, A., Lombardi, G., Ferrari, A. (2022). TinyCD: A (Not So) Deep Learning Model For Change Detection. Neural Computing and Applications. Published Online December, 18 2022. `https://link.springer.com/article/10.1007/s00521-022-08122-3` Reproduced with permission from Springer Nature. Code and datasets are available here: `https://github.com/AndreaCodegoni/Tiny_model_4_CD`

Chapter 2

Fourier Based Metrics

The best way to introduce the idea behind Optimal Transport is to follow the seminal work of Gaspar Monge, *Mémoire sur la théorie des déblais et des remblais* [34]. Suppose you want to build a sandcastle, and you have at your disposal a fixed amount of sand, stored in a pile with a specific shape. You have to rearrange the initial pile of sand into the location you have chosen for your sandcastle, and possibly in a form that resembles that of a castle. However, since you are on holiday, you want to do this job in an efficient way using the least amount of energy. The problem is: where do you transport every single grain of sand to build the castle using the least amount of energy? Another effective explanation of the Optimal Transport problem is the one proposed by Villani at the beginning of his book [35]: *"Consider a large number of bakeries, producing loaves, that should be transported each morning to cafés where consumers will eat them. The amount of bread that can be produced at each bakery and the amount that will be consumed at each café are known in advance, and can be modelled as probability measures (there is a "density of production" and a "density of consumption") on a certain space, which in our case would be Paris (equipped with the natural metric such that the distance between two points is the length of the shortest path joining them). The problem is to find in practice where each unit of bread should go, in such a way as to minimize the total transport cost."* These two stories highlight the essence of Optimal Transport: moving mass from an initial configuration to a target one, where moving is regulated by a cost which we want to keep low as possible.

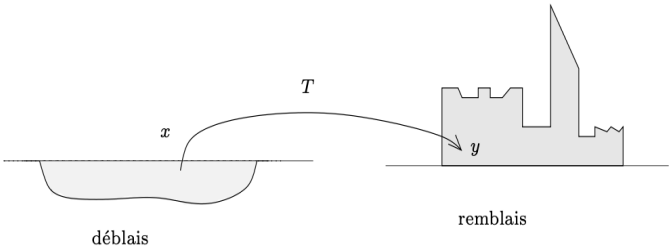

FIGURE 2.1: The déblais and remblais problem by Monge. Picture from [35]

Monge's initial version of the Optimal Transport is deterministic: a mass unit (a single grain of sand or a unit of bread) must be transported from the initial starting point to a specific target point. In other words, the Monge problem is an assignment problem [48]. At the beginning of 1900, this problem

was studied by a Russian mathematician, Leonid Vitaliyevich Kantorovich. In his works, he introduced the Duality theory in Linear Programming [49, 50], and then applied them to the original problem of Monge [51, 52]. One of the main aspects of Kantorovich's work was the introduction of the relaxed version of the Monge's assignment problem. Kantorovich admits that the mass can be broken, or in another interpretation, he assigns a destination to each quantity of mass with a certain probability. Monge, despite his great skills as a mathematician, had never taken care of relaxing the problem by removing the integrity constraints on the transported mass, probably because in applications this relaxation can lead to impractical situations. For example, what would be the point of carrying a seventy-second piece of a loaf from bakery A to cafés B. From an abstract point of view, however, this relaxation leads to a much more tractable problem. For example, for the relaxed problem, one can always prove the existence of a solution, which is not guaranteed in the case of the non-relaxed Monge problem [36]. Moreover, this change of paradigm allowed Kantorovich to define a notion of distance between probability measures [53]. Because of these fundamental contributions of Monge and Kantorovich, the problem of Optimal Transport is also known as Monge–Kantorovich problem. The distance between probability measures takes the name of Kantorovich–Rubinstein distance for this historical reason, but is widely known also with the name of Wasserstein distance [54], probably due to the terminology used in [55, 56]. In this work we have decided to use the name Wasserstein distance, but, as pointed out in [35,36], other names in different areas have also been attributed to this distance. Inspired by Monge's original problem, it is known as Earth Mover's Distance in the image processing community [57], in Statistic is known as Mallows distance [58] and as Tanaka distance in the field of partial differential equations [59, 60].

Another milestone in the history of Optimal Transport was set by George Dantzig. Dantzig, working in parallel with Kantorovich whose results were kept secret by the Soviet government, in his work [61], developed during the Second World War to cope with the logistics problems of the army, not only made important theoretical contributions to linear programming, but proposed and implemented the primal simplex algorithm. The primal simplex algorithm had a great impact on logistics problems, allowing it to automate the solution with great efficiency for the time it was developed and conceived, so much so that it earned its inventor the National Medal of Science. The applications of this algorithm go beyond logistical problems. In fact, the primal simplex algorithm can be used for all linear programming problems. In particular, the simplex algorithm represents the first numerical algorithm for the solution of the Monge-Kantorovich problem [62]. This follows from the fact that the Optimal Transport problem is equivalent, thanks to a result by Ford and Fulkerson [63], to an important class of linear programs problems known as minimum cost network flows [64].

In addition to the aforementioned logistic problems, the Wasserstein distance is used in statistics in problems concerning limit theorems and in all cases in which probability measures are to be compared [65–69]. In statistical mechanics, the Wasserstein distance is used to study the propagation of chaos and the

average behaviour of systems with many particles [70–73]. It is also useful for the study of Markov chains [74–76], for the asymptotic behaviour of partial differential equations [77–79], systems of particles [80].

In recent years, optimal transport has enjoyed enormous success in computer vision, where is used for image registration [81, 82], to transport/transfer the image style from a source image to a target one [83, 84], to construct classifiers that mimics the human eye perception [57, 85] and to compute barycenters among images [86, 87]. In computational biology, optimal transport shows the ability to take into account the relationships between different genes in order to classify cells by type or type of disease [88–90]. Also in machine learning, optimal transport has been used in generative models [91–93], in supervised learning [94], and in the context of domain adaptation, that is the task of transferring the knowledge from a well-known domain to another, less known and accessible domain [95, 96]. For the interested reader, that other applications of optimal transport can be found in [37, 97].

Given the large number of applications in which optimal transport is used, the interest in having efficient computational methods for solving these problems has grown. To speed up the simplex algorithm, in [98], the Optimal Transport problem is solved using a sequence of the shortest path problems on networks, while in [99] the authors exploit the structure of the problem to reduce the computational complexity. The advantage of this approach is that one can always attain the optimal value of the considered problem. In [100] the authors propose a tree-based algorithm for a particular case of Optimal Transport problem with quadratic complexity. Other researchers face the complexity of the LP problem by reducing the number of arcs in the flow formulation using truncated cost functions [85, 101]. The Sinkhorn algorithm [102] solves a regularized problem whose solution approximates the solution of the optimal transport, was recently brought to the attention of researchers in [103] where it was shown how this algorithm can be efficiently implemented as a matrix product on the GPU. Furthermore, unlike the linear programming problem, using the approximate problem, which makes us lose the optimal solution to the problem, allows us to have an objective function that can be easily differentiated through automatic differentiation algorithms.

In this chapter, guided by the need of efficient method linked to Optimal Transport, we present a family of metrics based on the Fourier transform, and we study the relationships between these metrics and the Wassertein distance. To this extent, we start in Section 2.1 recalling basic concepts of Optimal Transport more formally. In Section 2.2 we introduce the Fourier-based metrics. These types of metrics are widely used in kinetic theory, but, despite their equivalence with the Wasserstein metric, they have received little attention in the area of computer vision and signal processing. One of the possible causes is the requests to be made on moments of the distributions that are being compared to guarantee the finiteness of these metrics. To cope with this limitation in Section 2.3 we extend the classical Fourier-based metrics. Then, in Section 2.4 we study the extended Fourier-based metrics in a discrete setting and show explicitly the equivalence constants between Fourier-based metrics and Wasserstein distances.

To show the computational advantages of using Fourier-based metrics we fully describe their discrete form in Section 2.5 and in Section 2.6 we perform three different experiments: in the first and second experiment we numerically show that the results obtained with the Fourier-based metrics are numerically close or linearly correlatable to the results obtained with exact Optimal Transport solvers, but with several orders of magnitude more speed. Then in the last experiment, we show that the Fourier-based metrics could be also extended to compare generic vectors and not only probability measures with good performance in a classification task.

2.1 Background on Optimal Transport

In this section, we recall the basic definition and notions of Optimal Transport. A comprehensive introduction and theoretical advanced topics of Optimal Transport can be found in [35–37, 104, 105].

Let us start by fixing the notation. We work on the Euclidean space $\mathbb{R}^d$, endowed with the Borel $\sigma-$algebra $\mathcal{B}(\mathbb{R}^d)$. We use bold letters to denote vectors of $\mathbb{R}^d$. If $\boldsymbol{x} \in \mathbb{R}^d$, then x_i denotes its i-th coordinate. Given $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$, $\boldsymbol{x} \cdot \boldsymbol{y} = \sum_{i=1}^n x_i y_i$ is their scalar product and $|\boldsymbol{x}| = (\boldsymbol{x} \cdot \boldsymbol{x})^{1/2}$ is the Euclidean norm (or modulus) of $\boldsymbol{x}$. The set of probability measures on $\mathbb{R}^d$ is denoted by $\mathcal{P}(\mathbb{R}^d)$. Given $\mu \in \mathcal{P}(\mathbb{R}^d)$ and a Borel map $f : \mathbb{R}^d \to \mathbb{R}^d$, then the image measure (or push-forward) of μ by f is $f_{\#}\mu \in \mathcal{P}(\mathbb{R}^d)$, given by $f_{\#}\mu(A) = \mu(f^{-1}(A))$ for all $A \in \mathcal{B}(\mathbb{R}^d)$. Equivalently, for every continuous compactly supported function ϕ on $\mathbb{R}^d$, it holds

$$\int_{\mathbb{R}^d} \phi(y)\, d(f_{\#}\mu)(\boldsymbol{y}) = \int_{\mathbb{R}^d} \phi(f(x))\, d\mu(\boldsymbol{x}).$$

In order to make a formal definition of the Monge-Kantorovich problem, we need to introduce the concept of *transport plan.*

Definition 1 (Transport plan)**.** *Given two probability measures $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, a measure $\pi \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d)$ is called a transport plan between μ and ν if its marginals coincide with μ and ν, that is*

$$\pi(A \times \mathbb{R}^d) = \mu(A) \qquad \forall A \in \mathcal{B}(\mathbb{R}^d), \tag{2.1.1}$$
$$\pi(\mathbb{R}^d \times B) = \nu(B) \qquad \forall B \in \mathcal{B}(\mathbb{R}^d). \tag{2.1.2}$$

We denote by $\Pi(\mu, \nu)$ the set of all transport plans between μ and ν.

Now we can define the Monge-Kantorovich problem:

Definition 2 (Monge-Kantorovich problem)**.** *Given $\mu \in \mathcal{P}(\mathbb{R}^d)$, $\nu \in \mathcal{P}(\mathbb{R}^d)$ and $c : \mathbb{R}^d \times \mathbb{R}^d \to [0, +\infty]$, the Monge-Kantorovich problem is to find*

$$\inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} c(\boldsymbol{x}, \boldsymbol{y}) d\pi(\boldsymbol{x}, \boldsymbol{y}). \tag{2.1.3}$$

The first thing we need to worry about is the existence of a minimizer (2.1.3).

Theorem 1. *Given two probability distribution $\mu \in \mathcal{P}(\mathbb{R}^d)$ and $\nu \in \mathcal{P}(\mathbb{R}^d)$, and given a cost function $c : \mathbb{R}^d \times \mathbb{R}^d \to [0, +\infty]$, if c is lower semi-continuous, i.e.*

$$c(\boldsymbol{x}_0, \boldsymbol{y}_0) \leq \liminf_{(\boldsymbol{x},\boldsymbol{y})\to(\boldsymbol{x}_0,\boldsymbol{y}_0)} c(\boldsymbol{x}, \boldsymbol{y}), \quad \forall (\boldsymbol{x}_0, y_0) \in \mathbb{R}^d \times \mathbb{R}^d,$$

then the Monge-Kantorovich problem (2.1.3) *admits a solution and the* inf *is a minimum.*

Theorem 1 can be proved by using the direct method in the calculus of variation. The proof is out of the scope of this presentation and can be found in [35, Chapter 4] or [36, Chapter 1].

In the introduction to this chapter, the Monge-Kantorovich problem is presented as a linear programming problem. In fact, equation Equation (2) represents the objective function which is linear in π. Equation (2.1.1) and Equation (2.1.2) represent the constraints, which are linear with respect to π as they are marginalization. These two constraints means that the marginals of π must correspond to the two measures μ and ν. To make it even more explicit, we rewrite (2.1.3) and (2.1.1-2.1.2) in the case where μ and ν are two discrete probability measures. To this extent, suppose that μ is supported on n points $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots \boldsymbol{x}_n\}$ and ν on m points $\{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots \boldsymbol{y}_m\}$. This means that μ and ν are combination of Dirac delta functions, namely

$$\mu := \sum_{i=1}^{n} \mu_i \delta_{\boldsymbol{x}_i}, \quad \nu := \sum_{j=1}^{m} \nu_j \delta_{\boldsymbol{y}_j}. \tag{2.1.4}$$

In this case, the cost function can be identified as a matrix $c \in \mathbb{R}_+^{n\times m}$. The $c_{i,j}$ entry represent the cost of moving mass from source $\boldsymbol{x}_i$ to destination $\boldsymbol{y}_j$. The Monge-Kantorovich problem can then be written as

$$\min_{\pi\in\mathbb{R}_+^{n\times m}} \sum_{i=1}^{n}\sum_{j=1}^{m} c_{i,j}\pi_{i,j} \tag{2.1.5}$$

$$\text{s.t.} \sum_{j=0}^{m} \pi_{i,j} = \mu_i, \quad \forall i \in \{1, \cdots, n\} \tag{2.1.6}$$

$$\sum_{i=0}^{n} \pi_{i,j} = \nu_j, \quad \forall j \in \{1, \cdots, m\} \tag{2.1.7}$$

where $\pi_{i,j}$ represents the mass (to be determined) that flows form $\boldsymbol{x}_i$ to $\boldsymbol{y}_j$. We show in the discrete setting an example were the solution π of the Monge-Kantorovich problem is not unique.

Example 1. *Let us take as a support space for μ and ν the four vertex of a unit square in $\mathbb{R}^2$. We set*

$$\mu = \frac{1}{2}\delta_{[0,0]} + \frac{1}{2}\delta_{[1,1]}, \quad \nu = \frac{1}{2}\delta_{[1,0]} + \frac{1}{2}\delta_{[0,1]}.$$

As a cost function C we use the standard Euclidean distance in $\mathbb{R}^2$. In this case, we have

$$\pi^1 := \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}, \quad \pi^2 := \begin{bmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & 0 \end{bmatrix}$$

are both two optimal solutions to the Monge-Kantorovich problem with objective function equals to 1*. Moreover, any convex combination of π^1 and π^2 is an optimal solution. Hence, we have an infinite set of optimal solutions.*

The formal description of the Monge-Kantorovich problem allows us to understand why this problem has been so successful. (2.1.3) and its discrete counterpart (2.1.5), highlights how the formulation of the optimal transport problem can take into account the geometry of the problem. In fact, the cost function/matrix c can be chosen to adapt to the problem in question. For example, in the scenario proposed by Villani [35] on the distribution of goods, the cost matrix entries $c_{i,j}$ are the distances to be covered between bakeries and cafés. In [88], one of the proposed solutions defines the cost matrix using the correlations between genes, thus creating a matrix capable of taking into account the mutual behaviour of genes. These two possibilities, therefore, make the optimal transport formulation very flexible to tackle different types of problems and, in our opinion, this is the characteristic that has made it so popular in the application field.

It is worth noting that choosing properly the cost function c, leads to define a distance between probability measures. In fact, if for all $p \in \mathbb{N}$ we denote by $\mathcal{P}_p(\mathbb{R}^d)$ the set of probability measures with finite moments up to order p

$$\mathcal{P}_p(\mathbb{R}^d) := \left\{ \mu \in \mathcal{P}(\mathbb{R}^d) : \int_{\mathbb{R}^d} \boldsymbol{x}^\beta \, d\mu(\boldsymbol{x}) < +\infty, \ \forall \beta \in \mathbb{N}^d, \ |\beta| \leq p \right\},$$

and if we choose as cost function a distance $d : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, we can define the following distance.

Definition 3 (Wasserstein distance)**.** *Given $p \in \mathbb{N}$ and $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$, the Wasserstein distance of order p between μ and ν is defined as*

$$W_p(\mu, \nu) := \inf_{\pi \in \Pi(\mu,\nu)} \left\{ \int_{\mathbb{R}^d \times \mathbb{R}^d} d(\boldsymbol{x}, \boldsymbol{y})^p \, d\pi(\boldsymbol{x}, \boldsymbol{y}) \right\}^{1/p}, \tag{2.1.8}$$

where $d(\cdot, \cdot)$ is a distance on $\mathbb{R}^d$.

To check that Definition 3 is well-defined, we have to check the three axioms of distances:

Symmetry : since d is a distance and hence symmetric, also (2.1.8) is symmetric;

Degeneracy : d is a positive function and π is a positive measure and hence $W_p(\mu, \nu) \geq 0$ for all $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$. To see that $W_p(\mu, \nu) = 0$ if and only if $\mu = \nu$, we note that $d(\boldsymbol{x}, \boldsymbol{y}) = 0$ if and only if $\boldsymbol{x} = \boldsymbol{y}$. Thus, if $\mu = \nu$ no flow is needed and the optimal transport plan is concentrated on the diagonal

$\boldsymbol{x} = \boldsymbol{y}$ and we have $\int_{\mathbb{R}^d \times \mathbb{R}^d} 0^p \, d\pi(\boldsymbol{x}, \boldsymbol{y}) = 0$. For the other side, if π is not fully concentrated along the diagonal $\boldsymbol{x} = \boldsymbol{y}$, $\int_{\mathbb{R}^d \times \mathbb{R}^d} d(\boldsymbol{x}, \boldsymbol{y})^p \, d\pi(\boldsymbol{x}, \boldsymbol{y}) > 0$. So π must be non 0 only on $\boldsymbol{x} = \boldsymbol{y}$, and this means that there is no transport and hence $\mu = \nu$.

Triangular inequality : given $\mu, \nu, \zeta \in \mathcal{P}_p(\mathbb{R}^d)$ we have to show that $W_p(\mu, \nu) \leq W_p(\mu, \zeta) + W_p(\mu, \zeta)$. To prove this inequality we can use the Gluing Lemma. The Gluing Lemma and the proof of the triangular inequality can be found in [35]. Here we show how to proceed in the case of discrete measures. The Gluing Lemma says that one can construct a transport plan with a prescribed structure. In particular, we are interested in a transport plan obtained by gluing the optimal transport plan between μ and ζ, that we denote with $\pi^1 \in \mathbb{R}^d_+ \times \mathbb{R}^d_+$, and the optimal transport plan between ζ and ν, namely $\pi^2 \in \mathbb{R}^d_+ \times \mathbb{R}^d_+$. Now, to construct a transport plan $\pi^g \in \mathbb{R}^d_+ \times \mathbb{R}^d_+$ between μ and ν, we can glue π^1 and π^2 along their common marginal ζ. To this extent we define

$$\widetilde{\zeta}_i := \begin{cases} \zeta_i \text{ if } \zeta_i > 0 \\ 1 \text{ if } \zeta_i = 0, \end{cases}$$

and then we set

$$\pi^g := \pi^1 \mathrm{diag}(\frac{1}{\widetilde{\zeta}}) \pi^2 \in \mathbb{R}^d_+ \times \mathbb{R}^d_+,$$

where $\mathrm{diag}(\frac{1}{\widetilde{\zeta}})$ is the diagonal matrix with entries $\frac{1}{\widetilde{\zeta}}$. To verify that this is a transport plan between μ and ν, if we set $\mathbb{I}^d$ and $\mathbb{I}^{support(\zeta)}$ respectively as

$$\mathbb{I}^d_i := 1 \quad \forall i \in \{1, \ldots, d\}, \quad \mathbb{I}^{support(\zeta)}_i := \begin{cases} 1 \text{ if } \zeta_i > 0, \\ 0 \text{ if } \zeta_i = 0. \end{cases}$$

We can verify that

$$\pi^g \mathbb{I}^d = \pi^1 \mathrm{diag}(\frac{1}{\widetilde{\zeta}}) \pi^2 \mathbb{I}^d = \pi^1 \mathrm{diag}(\frac{\zeta}{\widetilde{\zeta}}) = \pi^1 \mathbb{I}^{support(\zeta)} = \mu$$

where the second inequality holds thanks to the fact that π^2 is an optimal transport plan between ζ and ν. Similarly, we obtain that $(\mathbb{I}^d)^T \pi^g = \nu$. Thus, π^g is a transport plan (not necessarily the optimal ones) between μ and ν. For simplicity, we suppose that i, j, k are all included in the range $\{1, \cdots, N\}$ which we will omit in order not to weigh down the notation.

Now, we are ready for the triangular inequality

$$W_p(\mu,\nu) = \left(\min_{\pi\in\Pi_{(\mu,\nu)}} \sum_{i,k} \pi_{i,k} d_{i,k}^p\right)^{1/p} \leq \left(\sum_{i,k} \pi_{i,k}^g d_{i,k}^p\right)^{1/p} \tag{2.1.9}$$

$$= \left(\sum_{i,k} d_{i,k}^p \sum_j \frac{\pi_{i,j}^1 \pi_{j,k}^2}{\widetilde{\zeta}_j}\right)^{1/p} \leq \left(\sum_{i,k,j} (d_{i,j}+d_{j,k})^p \frac{\pi_{i,j}^1 \pi_{j,k}^2}{\widetilde{\zeta}_j}\right)^{1/p} \tag{2.1.10}$$

$$\leq \left(\sum_{i,k,j} d_{i,j}^p \frac{\pi_{i,j}^1 \pi_{j,k}^2}{\widetilde{\zeta}_j}\right)^{1/p} + \left(\sum_{i,k,j} d_{j,k}^p \frac{\pi_{i,j}^1 \pi_{j,k}^2}{\widetilde{\zeta}_j}\right)^{1/p} \tag{2.1.11}$$

$$= \left(\sum_{i,j} d_{i,j}^p \pi_{i,j}^1 \sum_k \frac{\pi_{j,k}^2}{\widetilde{\zeta}_j}\right)^{1/p} + \left(\sum_{k,j} d_{j,k}^p \pi_{j,k}^2 \sum_i \frac{\pi_{i,j}^1}{\widetilde{\zeta}_j}\right)^{1/p} \tag{2.1.12}$$

$$= \left(\sum_{i,j} d_{i,j}^p \pi_{i,j}^1\right)^{1/p} + \left(\sum_{k,j} d_{j,k}^p \pi_{j,k}^2\right)^{1/p} \tag{2.1.13}$$

$$= W_p(\mu,\zeta) + W_p(\zeta,\nu),$$

where in (2.1.9) we have used the suboptimality of the transport plan π^g, in (2.1.10) the triangular inequality for the distance d, in (2.1.11) the Minkowski inequality, and to pass from (2.1.12) to (2.1.13) the marginalization of π^1 and π^2 and the definition of $\widehat{\zeta}$.

The Wasserstein distance (2.1.8) is not only of great interest in the application field, but is also important from a theoretical point of view. In fact, the following theorem [35, Theorem 6.9, p. 108] holds

Theorem 2. *Let $(\mu_k)_{k\in\mathbb{N}}$ a sequence of probability measures in $\mathcal{P}_m(\mathbb{R}^d)$ and $\mu \in \mathcal{P}_m(\mathbb{R}^d)$ another measure. Then, the two following statements are equivalent*

1. *μ_k converges weakly to μ, that is for all bounded and continuous function $\phi : \mathbb{R}^d \to \mathbb{R}$, we have that*

$$\int_{\mathbb{R}^d} \phi d\mu_k \to \int_{\mathbb{R}^d} \phi d\mu$$

2.

$$W_p(\mu_k,\mu) \to 0$$

The weak convergence plays a central role for example proving the existence of (weak) solution for partial differential equation [106–108]. However, this is not the only distance that parametrizes the weak convergence. Other examples are the Lévy-Prokhorov distance [109], the Fortet–Mourier distance [110] and the Toscani distance [35, Capter 6, p. 110]. Among others, Toscani's metrics caught our attention. As we will see in the next paragraphs, the equivalence between Toscani's metric and his generalizations, which we will call Fourier-based Metrics, will lead us to define equivalent metrics with the Wasserstein metric but which can be calculated much more efficiently.

In what follows, we focus on Wasserstein distances with exponents $p = 1$ and $p = 2$ and choose the Euclidean distance $d(\boldsymbol{x}, \boldsymbol{y}) := |\boldsymbol{x} - \boldsymbol{y}|$, namely

$$W_1(\mu, \nu) := \inf_{\pi \in \Pi(\mu,\nu)} \left\{ \int_{\mathbb{R}^d \times \mathbb{R}^d} |\boldsymbol{x} - \boldsymbol{y}| \, d\pi(\boldsymbol{x}, \boldsymbol{y}) \right\}, \tag{2.1.14}$$

$$W_2(\mu, \nu) := \inf_{\pi \in \Pi(\mu,\nu)} \left\{ \int_{\mathbb{R}^d \times \mathbb{R}^d} |\boldsymbol{x} - \boldsymbol{y}|^2 \, d\pi(\boldsymbol{x}, \boldsymbol{y}) \right\}^{1/2}. \tag{2.1.15}$$

2.2 Fourier-based metrics

In [40] Fourier-based metrics were used for the study of the trend to equilibrium for solutions of the spatially homogeneous Boltzmann equation for Maxwell molecules. Since then, many applications of these metrics have followed in both kinetic theory and probability [38, 39, 111–115]. All these problems deal with functions supported on the whole space $\mathbb{R}^d$, with $d \geq 1$, that exhibit a suitable decay at infinity which guarantees the existence of a suitable number of moments.

Given two probability measures $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, $d \geq 1$, and a real parameter $s > 0$, the Fourier-based metrics d_s considered in [40] are given by

$$d_s(\mu, \nu) := \sup_{\boldsymbol{k} \in \mathbb{R}^d \setminus \{0\}} \frac{|\widehat{\mu}(\boldsymbol{k}) - \widehat{\nu}(\boldsymbol{k})|}{|\boldsymbol{k}|^s}, \tag{2.2.16}$$

where $\widehat{\mu}$ and $\widehat{\nu}$ are the Fourier transforms of the measures μ and ν, respectively. As usual, given a probability measure $\mu \in \mathcal{P}(\mathbb{R}^d)$, the Fourier transform of μ is defined by

$$\hat{\mu}(\boldsymbol{k}) := \int_{\mathbb{R}^d} e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} d\mu(\boldsymbol{x}).$$

These metrics, for $s \geq 1$, are well-defined under the further assumption of boundedness and equality of moments of the probability measures. Indeed, a necessary condition for d_s to be finite, is that moments up to $\lfloor s \rfloor$ (the integer part of s) are equal for both measures [40].

In dimension $d = 1$, similar metrics were introduced a few years later by Baringhaus and Grübel in connection with the characterization of convex combinations of random variables [116]. Given two probability measures $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, $d \geq 1$, and two real parameters $s > 0$ and $p \geq 1$, the multi-dimensional version of these Fourier-based metrics reads

$$D_{s,p}(\mu, \nu) := \left(\int_{\mathbb{R}^d} \frac{|\widehat{\mu}(\boldsymbol{k}) - \widehat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{(ps+d)}} d\boldsymbol{k} \right)^{1/p}. \tag{2.2.17}$$

A limitation related to the application of the previous Fourier-based distances is related to its finiteness, which requires, for high values of s, a sufficiently high number of equal moments for the underlying probability measures.

Proposition 1 (Proposition 2.6, [39]). *Let $\lfloor s \rfloor$ denote the integer part of $s \in \mathbb{R}$ with $s \geq 1$, and assume that the densities $\mu, \nu \in \mathcal{P}_s(\mathbb{R}^d)$ possess equal moments up to $\lfloor s \rfloor$ if $s \notin \mathbb{N}$, or equal moments up to $s-1$ if $s \in \mathbb{N}$. Then the Fourier-based*

distance $d_s(\mu,\nu)$ is well-defined. In particular, $d_2(\mu,\nu)$ is well-defined for two densities with the same first moment.

In the context of kinetic equations of Boltzmann type, where conservation of momentum and energy of the solution is a consequence of the microscopic conservation laws of binary interactions among particles, this requirement on d_s, with $2 < s < 3$, is clearly not restrictive. However, to apply the Fourier-based metrics outside the context of kinetic equations, this requirement appears unnatural. To clarify this point, let us consider the case in which we want to compare the distance between two images. If we take two grayscale images and model them as probability distributions, there is no reason why these distributions possess the same expected value. The simplest example is given by two images consisting of a black dot, each one centred at a different point of the region, that can be modelled as two Dirac delta functions centred in two different points. A real-world example of two images with different mean values is reported in Figure 2.2.

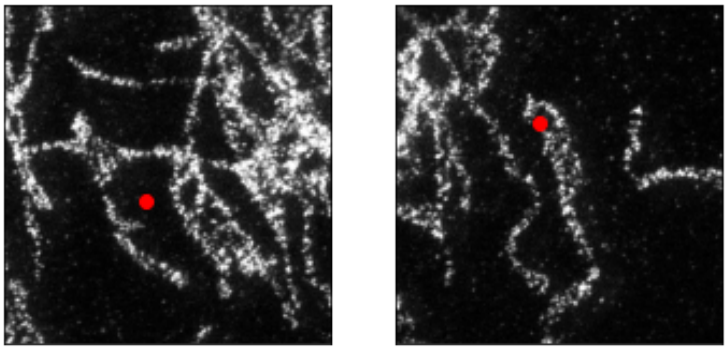

FIGURE 2.2: Two Microscopy images from [117] with their respective mean value highlighted with a red dot.

The interest in the d_2 metric is related to its equivalence to the Euclidean Wasserstein distance W_2. A detailed proof in dimension $d \geq 1$ can be found in the review paper [39].

Theorem 3 (Proposition 2.12 and Corollary 2.17, [39])**.** *For any given pair of probability densities $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$ such that $\int_{\mathbb{R}^d} \boldsymbol{x} d\mu(\boldsymbol{x}) = \int_{\mathbb{R}^d} \boldsymbol{x} d\nu(\boldsymbol{x})$, the d_2 metric is equivalent to the Euclidean Wasserstein distance W_2, that is, there exist two positive bounded constants $c < C$ such that*

$$cW_2(\mu,\nu) \leq d_2(\mu,\nu) \leq CW_2(\mu,\nu). \tag{2.2.18}$$

The proof in [39] does not provide in general the explicit expression of the two constants c and C. The value of these constants is quite involved, and it is strongly dependent on higher moments of the densities.

The lack of an explicit and tractable expression for the two equivalence constants of Theorem 3 could be a problem for the computational side. In fact, the quantification of these constants would tell us how much the value calculated with d_2 can be a good numerical approximation of the value calculated with W_2.

Despite the interesting and strong relation expressed in Theorem 3, which implies that d_2 and W_2 parametrize both the weak topology, we have no guarantees that the value computed with the two distances are the same or are well correlated. If the value is the same or if we have that $|c - C| \leq \epsilon$ with small $\epsilon \geq 0$, we could use the d_2 to compute an exact approximation of W_2. On the contrary, if we do not know the relation between c and C we have no information on how d_2 and W_2 are correlated numerically.

In the following sections, we firstly extend the definition of the Fourier Based metrics, and then we provide explicitly the two equivalence constant in a particular setting which is of interest for practical applications.

2.3 Extension of Fourier-based metrics

This section provides an extension of the Fourier-based metrics (2.2.16) for the case s = 2, which allows for a direct comparison between the Fourier-based metrics and the Wasserstein metric W_2.

In the previous section, we have seen that the Fourier-based metrics defined in (2.2.16) need some requirements about the moments of the probability measures we are considering. To overcome this limitation, we will make use of a property possessed by both the Fourier transform and the Wasserstein metric regarding the translations of the probability measures. Let us start by introducing the concept of the *center* of a distribution.

Definition 4 (Center of a distribution)**.** *Given $\mu \in \mathcal{P}_1(\mathbb{R}^d)$, we say that*

$$\boldsymbol{m}_\mu = \int_{\mathbb{R}^d} \boldsymbol{x} d\mu(\boldsymbol{x})$$

is the center *of μ.*

The center of a measure μ can be moved by resorting to a translation. Given $\mu \in \mathcal{P}_1(\mathbb{R}^d)$ and $\boldsymbol{\tau} \in \mathbb{R}^d$, we define the translated measure $\mu_{\boldsymbol{\tau}} \in \mathcal{P}_1(\mathbb{R}^d)$ by

$$\mu_{\boldsymbol{\tau}} = S^{\boldsymbol{\tau}}_{\#}\mu, \quad \text{where} \quad S^{\boldsymbol{\tau}}(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{\tau}.$$

Lemma 1. *Given $\mu, \nu \in \mathcal{P}_1(\mathbb{R}^d)$, there exists a unique vector $\boldsymbol{\tau} \in \mathbb{R}^d$ such that*

$$\boldsymbol{m}_\mu = \boldsymbol{m}_{\nu_{\boldsymbol{\tau}}}.$$

Proof. Let $\boldsymbol{\tau} := \boldsymbol{m}_\mu - \boldsymbol{m}_\nu$, then

$$\boldsymbol{m}_{\nu_{\boldsymbol{\tau}}} = \int_{\mathbb{R}^d} \boldsymbol{x} d\nu_{\boldsymbol{\tau}}(\boldsymbol{x}) = \int_{\mathbb{R}^d} (\boldsymbol{x} + \boldsymbol{\tau}) d\nu(\boldsymbol{x}) = \boldsymbol{m}_\nu + \boldsymbol{\tau} = \boldsymbol{m}_\mu.$$

□

Now we recall that the W_2 metric satisfies an explicit translation property [37, Remark 2.19]. We give below a short proof of this property.

Lemma 2. *Let $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$, with centers $\boldsymbol{m}_\mu$ and $\boldsymbol{m}_\nu$, respectively. For any given pair of vectors $\boldsymbol{v}, \boldsymbol{w} \in \mathbb{R}^d$ we have*

$$W_2(\mu_{\boldsymbol{v}}, \nu_{\boldsymbol{w}})^2 = W_2(\mu, \nu)^2 + |\boldsymbol{v} - \boldsymbol{w}|^2 + 2\langle \boldsymbol{v} - \boldsymbol{w}, \boldsymbol{m}_\mu - \boldsymbol{m}_\nu \rangle. \tag{2.3.19}$$

In addition, if we choose $\boldsymbol{v} = -\boldsymbol{m}_\mu$ and $\boldsymbol{w} = -\boldsymbol{m}_\nu$ it holds

$$W_2(\mu_{-\boldsymbol{m}_\mu}, \nu_{-\boldsymbol{m}_\nu})^2 = W_2(\mu, \nu)^2 - |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2. \tag{2.3.20}$$

Proof. Given a transport plan $\pi \in \Pi(\mu, \nu)$, we consider the transport plan

$$\tilde{\pi} := (S^{\boldsymbol{v}}, S^{\boldsymbol{w}})_{\#}\pi,$$

where $S^{\boldsymbol{v}}(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{v}$, $S^{\boldsymbol{w}}(\boldsymbol{y}) = \boldsymbol{y} + \boldsymbol{w}$. $\tilde{\pi}$ is a transport plan between the translated measures $\mu_{\boldsymbol{v}}$ and $\nu_{\boldsymbol{w}}$. Then, by definition of push-forward, we get

$$\begin{aligned}
&\int_{\mathbb{R}^d \times \mathbb{R}^d} |\boldsymbol{x} - \boldsymbol{y}|^2 d\tilde{\pi}(\boldsymbol{x}, \boldsymbol{y}) \\
&= \int_{\mathbb{R}^d \times \mathbb{R}^d} |(\boldsymbol{x} + \boldsymbol{v}) - (\boldsymbol{y} + \boldsymbol{w})|^2 d\pi(\boldsymbol{x}, \boldsymbol{y}) \\
&= \int_{\mathbb{R}^d \times \mathbb{R}^d} (|\boldsymbol{x} - \boldsymbol{y}|^2 + |\boldsymbol{v} - \boldsymbol{w}|^2 + 2\langle \boldsymbol{x} - \boldsymbol{y}, \boldsymbol{v} - \boldsymbol{w} \rangle) d\pi(\boldsymbol{x}, \boldsymbol{y}) \\
&= \int_{\mathbb{R}^d \times \mathbb{R}^d} |\boldsymbol{x} - \boldsymbol{y}|^2 d\pi(\boldsymbol{x}, \boldsymbol{y}) + |\boldsymbol{v} - \boldsymbol{w}|^2 + 2\langle \boldsymbol{m}_\mu - \boldsymbol{m}_\nu, \boldsymbol{v} - \boldsymbol{w} \rangle.
\end{aligned}$$

If π is an optimal transport plan between μ and ν, we have

$$\begin{aligned}
W_2(\mu_{\boldsymbol{v}}, \nu_{\boldsymbol{w}})^2 &\leq \int_{\mathbb{R}^d \times \mathbb{R}^d} |\boldsymbol{x} - \boldsymbol{y}|^2 d\tilde{\pi}(\boldsymbol{x}, \boldsymbol{y}) \\
&= W_2(\mu, \nu)^2 + |\boldsymbol{v} - \boldsymbol{w}|^2 + 2\langle \boldsymbol{v} - \boldsymbol{w}, \boldsymbol{m}_\mu - \boldsymbol{m}_\nu \rangle.
\end{aligned}$$

By repeating the previous argument with an optimal transport plan between $\mu_{\boldsymbol{v}},\ \nu_{\boldsymbol{w}}$, we find

$$\begin{aligned}
W_2(\mu_{\boldsymbol{v}}, \nu_{\boldsymbol{w}})^2 &= \int_{\mathbb{R}^d \times \mathbb{R}^d} |\boldsymbol{x} - \boldsymbol{y}|^2 d\pi(\boldsymbol{x}, \boldsymbol{y}) + |\boldsymbol{v} - \boldsymbol{w}|^2 + 2\langle \boldsymbol{v} - \boldsymbol{w}, \boldsymbol{m}_\mu - \boldsymbol{m}_\nu \rangle \\
&\geq W_2(\mu, \nu)^2 + |\boldsymbol{v} - \boldsymbol{w}|^2 + 2\langle \boldsymbol{v} - \boldsymbol{w}, \boldsymbol{m}_\mu - \boldsymbol{m}_\nu \rangle.
\end{aligned}$$

Hence, we can conclude

$$W_2(\mu_{\boldsymbol{v}}, \nu_{\boldsymbol{w}})^2 = W_2(\mu, \nu)^2 + |\boldsymbol{v} - \boldsymbol{w}|^2 + 2\langle \boldsymbol{v} - \boldsymbol{w}, \boldsymbol{m}_\mu - \boldsymbol{m}_\nu \rangle.$$

□

Following the property of Wasserstein distance W_2 stated in Lemma 2, we modify the Fourier-based metrics d_2 and $D_{2,p}$ in such a way to allow for probability measures with different centers of mass. We start by considering the case of the metric d_2.

Definition 5 (Translated Fourier-based Metric)**.** *We define the function* $\mathcal{D}_2 : \mathcal{P}_2(\mathbb{R}^d) \times \mathcal{P}_2(\mathbb{R}^d) \to \mathbb{R}$ *as:*

$$\mathcal{D}_2(\mu,\nu) := \sqrt{d_2(\mu,\nu_{\boldsymbol{m}_\mu-\boldsymbol{m}_\nu})^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2}. \tag{2.3.21}$$

Owing to Lemma 1 and Proposition 1, $\mathcal{D}_2(\mu,\nu)$ is well-defined for each pair of probability measures in $\mathcal{P}_2(\mathbb{R}^d)$, independently of their centers. Note that $\nu_{\boldsymbol{m}_\mu-\boldsymbol{m}_\nu}$, which is the translation of ν by $\boldsymbol{m}_\mu - \boldsymbol{m}_\nu$, has the same center as μ. One could give an equivalent definition of $\mathcal{D}_2$ by translating μ, instead of ν, or by translating both centers to $\mathbf{0}$.

Lemma 3. *Given* $\mu,\nu \in \mathcal{P}_2(\mathbb{R}^d)$ *and* $\boldsymbol{v},\boldsymbol{w} \in \mathbb{R}^d$*, then*

$$|\widehat{\mu_{\boldsymbol{v}}}(\boldsymbol{k}) - \widehat{\nu_{\boldsymbol{w}}}(\boldsymbol{k})| = |\hat{\mu}(\boldsymbol{k}) - \widehat{\nu_{\boldsymbol{w}-\boldsymbol{v}}}(\boldsymbol{k})| = |\widehat{\mu_{\boldsymbol{v}-\boldsymbol{w}}}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|.$$

Therefore,

$$d_2(\mu_{\boldsymbol{v}},\nu_{\boldsymbol{w}}) = d_2(\mu,\nu_{\boldsymbol{w}-\boldsymbol{v}}) = d_2(\mu_{\boldsymbol{v}-\boldsymbol{w}},\nu).$$

In particular, the function $(\mu,\nu) \to d_2(\mu,\nu_{\boldsymbol{m}_\mu-\boldsymbol{m}_\nu})$ *is symmetric.*

Proof. By the translation property of the Fourier Transform, for all $\boldsymbol{v} \in \mathbb{R}^d$ we have the identity

$$\widehat{\mu_{\boldsymbol{v}}}(\boldsymbol{k}) = e^{-i\boldsymbol{v}\cdot\boldsymbol{k}}\hat{\mu}(\boldsymbol{k}).$$

Therefore,

$$\begin{aligned} |e^{-i\boldsymbol{v}\cdot\boldsymbol{k}}\hat{\mu}(\boldsymbol{k}) - e^{-i\boldsymbol{w}\cdot\boldsymbol{k}}\hat{\nu}(\boldsymbol{k})| &= |e^{-i\boldsymbol{w}\cdot\boldsymbol{k}}(e^{-i(\boldsymbol{v}-\boldsymbol{w})\cdot\boldsymbol{k}}\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k}))| \\ &= |e^{-i(\boldsymbol{v}-\boldsymbol{w})\cdot\boldsymbol{k}}\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|. \end{aligned}$$

This shows that

$$\sup_{\boldsymbol{k}\in\mathbb{R}^d\setminus\{0\}} \frac{|e^{-i\boldsymbol{v}\cdot\boldsymbol{k}}\hat{\mu}(\boldsymbol{k}) - e^{-i\boldsymbol{w}\cdot\boldsymbol{k}}\hat{\nu}(\boldsymbol{k})|}{|\boldsymbol{k}|^2} = \sup_{\boldsymbol{k}\in\mathbb{R}^d\setminus\{0\}} \frac{|e^{-i(\boldsymbol{v}-\boldsymbol{w})\cdot\boldsymbol{k}}\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|}{|\boldsymbol{k}|^2}.$$

□

Lemma 3 implies the following theorem.

Theorem 4. *The function* $\mathcal{D}_2$ *defined in* (2.3.21) *is a distance over* $\mathcal{P}_2(\mathbb{R}^d)$*.*

Proof. Clearly $\mathcal{D}_2(\mu,\nu) \geq 0, \forall \mu,\nu \in \mathcal{P}_2(\mathbb{R}^d)$, and $\mathcal{D}_2(\mu,\nu) = 0$ if and only if $\mu = \nu$. Symmetry follows from Lemma 3. Finally, in reason of the fact that both $d_2(\mu,\nu)$ and $|\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|$ are distances, $\mathcal{D}_2$ satisfies the triangular inequality. □

An analogous extension can be done for the metric $D_{2,p}$ defined in (2.2.17).

Definition 6. *Given* $p \geq 1$*, we define* $\mathcal{D}_{2,p} : \mathcal{P}_2(\mathbb{R}^d) \times \mathcal{P}_2(\mathbb{R}^d) \to \mathbb{R}$ *by*

$$\mathcal{D}_{2,p}(\mu,\nu) := \sqrt{D_{2,p}(\mu,\nu_{\boldsymbol{m}_\mu-\boldsymbol{m}_\nu})^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2}.$$

$\mathcal{D}_{2,p}$ *is a metric on* $\mathcal{P}_2(\mathbb{R}^d)$*.*

Using Definition 5 of $\mathcal{D}_2$ metric, we are able to extend the result of Theorem 3 also to probability measures with different centers:

Theorem 5. *The function $\mathcal{D}_2$ defined in* (2.3.21) *is equivalent to the W_2 distance.*

Proof. Let $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$ and let μ^*, ν^* denote the two corresponding translated measures centered in $\mathbf{0}$. By Lemma 2, we have

$$W_2^2(\mu, \nu) = W_2^2(\mu^*, \nu^*) + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2. \tag{2.3.22}$$

Owing to Theorem 3, there exist two constants $c, C \in (0, \infty)$ such that

$$c d_2(\mu^*, \nu^*) \leq W_2(\mu^*, \nu^*) \leq C d_2(\mu^*, \nu^*). \tag{2.3.23}$$

Using (2.3.22) in (2.3.23), we get

$$c d_2(\mu^*, \nu^*)^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2 \leq W_2(\mu, \nu)^2 \leq C d_2(\mu^*, \nu^*)^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2,$$

which can be rewritten as

$$\begin{aligned} \min\{c, 1\}\big(d_2(\mu^*, \nu^*)^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2\big) &\leq W_2(\mu, \nu)^2 \\ &\leq \max\{1, C\}\big(d_2(\mu^*, \nu^*)^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2\big). \end{aligned}$$

Finally,

$$\min\{c, 1\}\, \mathcal{D}_2^2(\mu, \nu) \leq W_2^2(\mu, \nu) \leq \max\{1, C\}\, \mathcal{D}_2^2(\mu, \nu).$$

□

2.4 The Periodic Fourier-based metrics

In this section, we introduce a family of (Discrete) Periodic Fourier-based metrics suitable to measure the distance between discrete probability measures whose support is restricted to a given set of points, and we discuss their equivalence with the Wasserstein metrics. The main result is that in this case, one obtains a precise estimation of the constants of equivalence.

Remark 1. *When the space on which the measures are supported has finite measure, talking about equivalence may seem trivial. In fact, as long as the total measure of the support is included in the equivalence constants if necessary, all distances are equivalent. However, the type of equivalence, and therefore how the metrics behave, depends on whether the total measure of the space is present in the equivalence constants or not. We will return to this observation later.*

Definition 7 (Regular grid)**.** *For $N \in \mathbb{N} \setminus \{0\}$, we define the mono dimensional regular grid*

$$G_N := \big\{0, \frac{1}{N}, \dots, \frac{N-1}{N}\big\}$$

Note that $G_N \subset [0, 1)$. The 2-dimensional regular grid $G_N^2 \subset [0, 1)^2$ is defined through the Cartesian product $G_N \times G_N$, and similarly we can define G_N^d, the d-dimensional regular grid.

Definition 8 (Discrete Measure over a grid)**.** *We say that μ is a discrete measure over G_N^d if its support is contained in G_N^d, that is, if μ has the form*

$$\mu(\boldsymbol{x}) = \sum_{\boldsymbol{y}\in G_N^d} \mu_{\boldsymbol{y}} \delta_0(\boldsymbol{x}-\boldsymbol{y}), \tag{2.4.24}$$

where $\mu_{\boldsymbol{y}} \in \mathbb{R}, \mu_{\boldsymbol{y}} \geq 0$ for all $\boldsymbol{y} \in G_N^d$. The Discrete Fourier transform of a discrete measure over G_N^d is given by

$$\hat{\mu}(\boldsymbol{k}) = \sum_{\boldsymbol{x}\in G_N^d} \mu_{\boldsymbol{x}} e^{-i\boldsymbol{x}\cdot\boldsymbol{k}}. \tag{2.4.25}$$

The periodicity of the complex exponential implies that $\hat{\mu}$ is $2\pi N$-periodic over all directions, so that it is sufficient to study $\hat{\mu}$ over a strict subset of $\mathbb{R}^d$, e.g., over $[0, 2\pi N]^d$. For instance, the value of the Fourier-based metric (2.2.16) is achieved by searching for the "sup" operator on the bounded set $[0, 2\pi N]^d$. Since

$$\frac{1}{|\boldsymbol{k}|^2} \geq \frac{1}{|\boldsymbol{k}'|^2}, \quad \forall \boldsymbol{k} \in (0, 2\pi N]^d, \, \forall \boldsymbol{k}' \in \mathbb{R}_+^d \backslash [0, 2\pi N]^d$$

and the function

$$\boldsymbol{k} \to |\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|$$

is $2\pi N$-periodic, for any given constant $s > 0$ the Discrete Fourier-based metric can be defined as

$$d_s(\mu, \nu) = \sup_{\boldsymbol{k}\in[0,2\pi N]^d\backslash\{0\}} \frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|}{|\boldsymbol{k}|^s}. \tag{2.4.26}$$

Definition 9 (Dilated Discrete Measures)**.** *Given a discrete measure μ over G_N^d and $\gamma \in \mathbb{R}$ such that $\gamma > 0$, the γ-dilated measure μ_γ is*

$$\mu_\gamma(\boldsymbol{x}) = \sum_{\boldsymbol{y}\in G_N^d} \mu_{\boldsymbol{y}} \delta_0(\gamma\boldsymbol{x} - \boldsymbol{y}).$$

The Fourier transform of μ_γ is

$$\hat{\mu}_\gamma(\boldsymbol{k}) = \sum_{\boldsymbol{x}\in G_N^d} \mu_{\boldsymbol{x}} e^{-\frac{i}{\gamma}\langle\boldsymbol{k},\boldsymbol{x}\rangle} = \hat{\mu}\left(\frac{\boldsymbol{k}}{\gamma}\right). \tag{2.4.27}$$

Therefore, if $\hat{\mu}$ is T-periodic, then $\hat{\mu}_\gamma$ is γT-periodic. Like the original metrics (2.2.16)*, the metric (2.4.26) satisfies the dilation property*

$$d_s(\mu_\gamma, \nu_\gamma) = \frac{1}{\gamma^s} d_s(\mu, \nu). \tag{2.4.28}$$

In particular, if we consider μ of the form (2.4.24), the Fourier transform of its $\frac{1}{N}$-dilation is 2π-periodic.

We recall the definition of the metrics (2.2.17):

$$D_{s,p}(\mu,\nu) := \left(\int_{\mathbb{R}^d} \frac{|\widehat{\mu}(\boldsymbol{k}) - \widehat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{(sp+d)}} d\boldsymbol{k} \right)^{\frac{1}{p}},$$

where $s > 0$ and $p \geq 1$. As we did for the Fourier-Based Metrics d_s, thanks to the periodicity of the Fourier transform, we can restrict the domain of integration to $[0,T]^d$. In this case, for any given choice of the parameters p and s, this distance is well-defined any time the integrand is integrable in a neighbourhood of the origin. This corresponds to requiring that $\frac{1}{|\boldsymbol{k}|^\gamma}$ is integrable on the d-dimensional ball $B_1(0) = \{\boldsymbol{k} \in \mathbb{R}^d : |\boldsymbol{k}| \leq 1\}$, that is, if and only if $\gamma < d$. This consideration suggests the following definition.

Definition 10 (The Periodic Fourier-based metric)**.** *Let μ and ν be two probability measures over G_N^d. The (s,p,α)-Periodic Fourier-based metric between μ and ν is defined as*

$$f_{s,p}^{(\alpha)}(\mu,\nu) := \left(\frac{1}{|T|^d} \int_{[0,T]^d} \frac{|\widehat{\mu}(\boldsymbol{k}) - \widehat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{sp+\alpha}} d\boldsymbol{k} \right)^{\frac{1}{p}}, \tag{2.4.29}$$

where $p, s, \alpha \in \mathbb{R}$ and T is the period of $\hat{\mu}$ and $\hat{\nu}$. When $\alpha = 0$ and $s \in \mathbb{N}$ we say that $f_{s,p} := f_{s,p}^{(0)}$ is pure.

As discussed in Section 2.2, in dimension $d = 1$ the continuous version of the metrics (2.4.29) has been considered in [116]. Recently, these metrics have been considered in relation to the problem of convergence toward equilibrium of a Fokker–Planck type equation modelling wealth distribution [118], where various properties of these metrics have been studied. As pointed out in [118], if μ and ν have equal r-moments, the function $|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|$ behaves like $|\boldsymbol{k}|^{r+1}$ as $\boldsymbol{k} \to 0$. As a consequence, the value of $f_{s,p}^{(\alpha)}(\mu,\nu)$ is finite only if the following condition is verified

$$p(s - r - 1) + \alpha < d. \tag{2.4.30}$$

If s, p and α satisfy (2.4.30), and thus $f_{s,p}^{(\alpha)} < +\infty$, we say that $f_{s,p}^{(\alpha)}$ is feasible.

Proposition 2. *Let μ and ν be two probability measures over G_N^d. For any given constant $\gamma > 0$, the following dilation property holds*

$$f_{s,p}^{(\alpha)}(\mu_\gamma, \nu_\gamma) = \frac{1}{|\gamma|^{s+\frac{\alpha}{p}}} f_{s,p}^{(\alpha)}(\mu,\nu).$$

Proof. Using relation (2.4.27) and the change of variables $\boldsymbol{k} = \gamma \boldsymbol{k}'$, we get

$$\begin{aligned} f_{s,p}^{(\alpha)}(\mu_\gamma, \nu_\gamma) &= \left(\frac{1}{|\gamma T|^d} \int_{[0,\gamma T]^d} \frac{|\hat{\mu}_\gamma(\boldsymbol{k}) - \hat{\nu}_\gamma(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{sp+\alpha}} d\boldsymbol{k} \right)^{\frac{1}{p}} \\ &= \left(\frac{1}{|\gamma T|^d} \int_{[0,\gamma T]^d} \frac{|\hat{\mu}(\frac{\boldsymbol{k}}{\gamma}) - \hat{\nu}(\frac{\boldsymbol{k}}{\gamma})|^p}{|\boldsymbol{k}|^{sp+\alpha}} d\boldsymbol{k} \right)^{\frac{1}{p}} \\ &= \left(\frac{1}{|\gamma|^d} \frac{1}{|T|^d} \int_{[0,T]^d} \frac{|\hat{\mu}(\boldsymbol{k}') - \hat{\nu}(\boldsymbol{k}')|^p}{|\gamma|^{sp+\alpha} |\boldsymbol{k}'|^{sp+\alpha}} |\gamma|^d d\boldsymbol{k}' \right)^{\frac{1}{p}} \\ &= \frac{1}{|\gamma|^{s+\frac{\alpha}{p}}} \left(\frac{1}{|T|^d} \int_{[0,T]^d} \frac{|\hat{\mu}(\boldsymbol{k}') - \hat{\nu}(\boldsymbol{k}')|^p}{|\boldsymbol{k}'|^{sp+\alpha}} d\boldsymbol{k}' \right)^{\frac{1}{p}} \\ &= \frac{1}{|\gamma|^{s+\frac{\alpha}{p}}} f_{s,p}^{(\alpha)}(\mu, \nu). \end{aligned}$$

□

It is important to remark that, differently from the metrics (2.2.17), the analogous of the dilation property (2.4.28) is true only for $\alpha = 0$, that is only for pure metrics. We show next that the $f_{s,p}^{(\alpha)}$ metrics satisfy various monotonicity properties with respect to the parameters p and s.

Proposition 3. *Let μ and ν be two probability measures over G_N^d, with moments equal up to r. If $t \leq s$, then*

$$f_{t,p}^{(\alpha)}(\mu, \nu) \leq (\sqrt{d}|T|)^{(s-t)} f_{s,p}^{(\alpha)}(\mu, \nu),$$

for any p and α for which the metric is feasible, i.e., for $p(s - r - 1) + \alpha < d$.

Proof. We compute

$$\begin{aligned} f_{t,p}^{(\alpha)}(\mu, \nu) &= \left(\frac{1}{|T|^d} \int_{[0,T]^d} \frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{tp+\alpha}} d\boldsymbol{k} \right)^{\frac{1}{p}} \\ &= \left(\frac{1}{|T|^d} \int_{[0,T]^d} \frac{|\boldsymbol{k}|^{p(s-t)}}{|\boldsymbol{k}|^{p(s-t)}} \frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{tp+\alpha}} d\boldsymbol{k} \right)^{\frac{1}{p}} \\ &= \left(\frac{1}{|T|^d} \int_{[0,T]^d} |\boldsymbol{k}|^{p(s-t)} \frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{sp+\alpha}} d\boldsymbol{k} \right)^{\frac{1}{p}} \\ &\leq (\sqrt{d}|T|)^{(s-t)} f_{s,p}^{(\alpha)}(\mu, \nu). \end{aligned}$$

The last inequality is obtained resorting to the bound $|\boldsymbol{k}| \leq \sqrt{d}|T|$. □

Proposition 4. *Let μ and ν be two probability measures over G_N^d. If $\alpha = 0$ and $p \leq q$, then*

$$f_{s,p}(\mu, \nu) \leq f_{s,q}(\mu, \nu).$$

Proof. We have

$$\begin{aligned} f_{s,p}(\mu,\nu) &= \left(\frac{1}{|T|^d}\int_{[0,T]^d}\frac{|\hat{\mu}(\boldsymbol{k})-\hat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{sp}}d\boldsymbol{k}\right)^{\frac{1}{p}} \\ &= \left(\left(\frac{1}{|T|^d}\int_{[0,T]^d}\frac{|\hat{\mu}(\boldsymbol{k})-\hat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{sp}}d\boldsymbol{k}\right)^{\frac{q}{p}}\right)^{\frac{1}{q}} \\ &\leq \left(\frac{1}{|T|^d}\int_{[0,T]^d}\left(\frac{|\hat{\mu}(\boldsymbol{k})-\hat{\nu}(\boldsymbol{k})|^p}{|\boldsymbol{k}|^{sp}}\right)^{\frac{q}{p}}d\boldsymbol{k}\right)^{\frac{1}{q}} \\ &= f_{s,q}(\mu,\nu). \end{aligned}$$

The last inequality follows from Jensen's inequality. □

Remark 2. *By letting $p \to +\infty$, we get*

$$\lim_{p\to\infty} f_{s,p}(\mu,\nu) = f_{s,\infty}(\mu,\nu) := d_s(\mu,\nu).$$

Thanks to the Hölder inequality, for all $p < +\infty$ we have the bound

$$f_{s,p}(\mu,\nu) \leq d_s(\mu,\nu). \tag{2.4.31}$$

The results of this subsection are preliminary to our main result, which deals with the equivalence of the pure metrics, for $p = 2$, with the Wasserstein metrics. For the sake of simplicity, and without loss of generality, in the next subsection we consider measures in dimension $d = 2$.

2.4.1 Equivalence with the Wasserstein metric W_1

We consider the two cases $s = 1$ and $s = 2$, in dimension $d = 2$, and we show that $f_{1,2}$ and $f_{2,2}$ are equivalent to W_1 and W_2, respectively.

We start with the case $s = 1$. For any $\mu, \nu \in \mathcal{P}(G_N^d)$, the Periodic Fourier-based metrics is

$$f_{1,2}(\mu,\nu) = \left(\frac{1}{|T|^2}\int_{[0,T]^2}\frac{|\hat{\mu}(\boldsymbol{k})-\hat{\nu}(\boldsymbol{k})|^2}{|\boldsymbol{k}|^2}d\boldsymbol{k}\right)^{\frac{1}{2}}. \tag{2.4.32}$$

We have the following

Theorem 6. *For any pair of measures $\mu, \nu \in \mathcal{P}(G_N^d)$, we have the inequality*

$$f_{1,2}(\mu,\nu) \leq W_1(\mu,\nu).$$

Proof. Let π be a transport plan between μ and ν. It holds

$$\begin{aligned}
|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})| &= \Big| \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} \pi(\boldsymbol{x},\boldsymbol{y}) - \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} e^{-i\boldsymbol{k}\cdot\boldsymbol{y}} \pi(\boldsymbol{x},\boldsymbol{y}) \Big| \\
&= \Big| \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} \big(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} - e^{-i\boldsymbol{k}\cdot\boldsymbol{y}}\big) \pi(\boldsymbol{x},\boldsymbol{y}) \Big| \\
&\leq \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} \big|e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} - e^{-i\boldsymbol{k}\cdot\boldsymbol{y}}\big| \pi(\boldsymbol{x},\boldsymbol{y}) \\
&= \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} \big|1 - e^{i\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{y})}\big| \pi(\boldsymbol{x},\boldsymbol{y}) \\
&\leq \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} \big|\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{y})\big| \pi(\boldsymbol{x},\boldsymbol{y}) \\
&\leq |\boldsymbol{k}| \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} |\boldsymbol{x}-\boldsymbol{y}| \pi(\boldsymbol{x},\boldsymbol{y}).
\end{aligned}$$

Hence, if π is the optimal transport plan, we conclude with the inequality

$$|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})| \leq |\boldsymbol{k}| W_1(\mu,\nu). \tag{2.4.33}$$

Using inequality (2.4.33) into definition (2.4.32), we finally obtain the bound

$$f_{1,2}(\mu,\nu) \leq \left(\frac{1}{|T|^2} \int_{[0,T]^2} \frac{\big(|\boldsymbol{k}| W_1(\mu,\nu)\big)^2}{|\boldsymbol{k}|^2} d\boldsymbol{k} \right)^{\frac{1}{2}} = W_1(\mu,\nu). \tag{2.4.34}$$

□

Since $W_1(\mu,\nu) < +\infty$ for every $\mu,\nu \in \mathcal{P}(G_N^d)$, inequality (2.4.34) implies that $f_{1,2}$ is bounded in correspondence to any pair of probability measures over the grid G_N^d.

We now show that $f_{1,2}$ and W_1 satisfy a reverse inequality, thus concluding that the two metrics are equivalent.

Theorem 7. *For any pair of measures $\mu,\nu \in \mathcal{P}(G_N^d)$ it holds*

$$W_1(\mu,\nu) \leq \frac{T^2}{2\pi} f_{1,2}(\mu,\nu). \tag{2.4.35}$$

Proof. Owing to the dual characterization of the W_1 distance (see [35], Chapter 5), there exists a 1-Lipschitz function ϕ such that

$$W_1(\mu,\nu) = \int_{\mathbb{R}^2} \phi(\boldsymbol{x}) d\mu(\boldsymbol{x}) - \int_{\mathbb{R}^2} \phi(\boldsymbol{x}) d\nu(\boldsymbol{x}).$$

Since μ and ν are discrete and supported on a subset of $[0,1]^2$, we can write

$$W_1(\mu,\nu) = \sum_{\boldsymbol{x} \in G_N^d} \phi(\boldsymbol{x}) \big(\mu_{\boldsymbol{x}} - \nu_{\boldsymbol{x}}\big).$$

Therefore, resorting to the fact that both the measures have the same mass, for any given constant $c \in \mathbb{R}$ we have

$$W_1(\mu,\nu) = \sum_{\boldsymbol{x}\in G_N^d} \big(\phi(\boldsymbol{x}) + c\big)\big(\mu_{\boldsymbol{x}} - \nu_{\boldsymbol{x}}\big).$$

The last identity permits choosing ϕ such that $\phi(\frac{N}{2},\frac{N}{2}) = 0$. Since ϕ is 1-Lipschitz, we conclude that

$$|\phi(\boldsymbol{x})| \le \frac{\sqrt{2}}{2}, \qquad \forall \boldsymbol{x} \in G_N^d. \tag{2.4.36}$$

By the Hölder inequality we obtain

$$W_1(\mu,\nu) \le \Big(\sum_{\boldsymbol{x}\in G_N^d} |\phi(\boldsymbol{x})|^2\Big)^{\frac{1}{2}} \Big(\sum_{\boldsymbol{x}\in G_N^d} |\mu_{\boldsymbol{x}} - \nu_{\boldsymbol{x}}|^2\Big)^{\frac{1}{2}}.$$

Since

$$\sum_{\boldsymbol{x}\in G_N^d} |\mu_{\boldsymbol{x}} - \nu_{\boldsymbol{x}}|^2 = \frac{1}{|T|^2}\int_{[0,T]^2} A(\boldsymbol{k})B(\boldsymbol{k})d\boldsymbol{k}$$

where

$$\begin{aligned} A(\boldsymbol{k}) &= \sum_{\boldsymbol{x}\in G_N^d} (\mu_{\boldsymbol{x}} - \nu_{\boldsymbol{x}})e^{-i<\boldsymbol{x},\boldsymbol{k}>} \\ B(\boldsymbol{k}) &= \sum_{\boldsymbol{y}\in G_N^d} (\mu_{\boldsymbol{y}} - \nu_{\boldsymbol{y}})e^{+i<\boldsymbol{y},\boldsymbol{k}>} \end{aligned}$$

we have

$$\sum_{\boldsymbol{x}\in G_N^d} |\mu_{\boldsymbol{x}} - \nu_{\boldsymbol{x}}|^2 = \frac{1}{|T|^2}\int_{[0,T]^2} \big|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})\big|^2 d\boldsymbol{k}.$$

Now using (2.4.36) we obtain

$$\begin{aligned} W_1(\mu,\nu) &\le \frac{\sqrt{2}N}{2}\Big(\frac{1}{|T|^2}\int_{[0,T]^2} \big|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})\big|^2 d\boldsymbol{k}\Big)^{\frac{1}{2}} \\ &= \frac{\sqrt{2}N}{2}\Big(\frac{1}{|T|^2}\int_{[0,T]^2} |\boldsymbol{k}|^2 \frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^2}{|\boldsymbol{k}|^2} d\boldsymbol{k}\Big)^{\frac{1}{2}}. \end{aligned}$$

Since $|\boldsymbol{k}|^2 \le 2T^2$ and $T = 2\pi N$, we can finally conclude that

$$W_1(\mu,\nu) \le \frac{T^2}{2\pi}\Big(\frac{1}{|T|^2}\int_{[0,T]^2} \frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^2}{|\boldsymbol{k}|^2} d\boldsymbol{k}\Big)^{\frac{1}{2}} = \frac{T^2}{2\pi} f_{1,2}(\mu,\nu).$$

□

In consequence of the previous estimates, it is immediate to show that the metrics d_s and W_1 are equivalent. This is proven in the following

Corollary 1. *For any pair of measures* $\mu, \nu \in \mathcal{P}(G_N^d)$

$$d_1(\mu,\nu) \leq W_1(\mu,\nu) \leq \frac{T^2}{2\pi} d_1(\mu,\nu).$$

Proof. The first inequality is a consequence of bound (2.4.33). The second one follows from inequality (2.4.31). □

2.4.2 Equivalence with the Wasserstein metric W_2

The aim of this Section is to show the equivalence of the Fourier-based metric $f_{2,2}$ and the Wasserstein metric W_2. Let $s = 2$. In this case, the Periodic Fourier-based metrics takes the form

$$f_{2,2}(\mu,\nu) = \left(\frac{1}{|T|^2} \int_{[0,T]^2} \frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^2}{|\boldsymbol{k}|^4} d\boldsymbol{k} \right)^{\frac{1}{2}}. \tag{2.4.37}$$

The distance between the two probability measures is well-defined only when μ and ν possess the same expected value. Since, in general, this is not the case, we start by translating the measures, as done in Section 2.3, to satisfy this condition. The following proposition shows that, for probability measures with the same centre, the topology induced by $f_{2,2}$ is not stronger than the topology induced by W_2.

Theorem 8. *For any pair of measures* $\mu, \nu \in \mathcal{P}(G_N^d)$ *such that* $\boldsymbol{m}_\mu = \boldsymbol{m}_\nu$*, it holds*

$$f_{2,2}(\mu,\nu) \leq 2\sqrt{2} W_2(\mu,\nu). \tag{2.4.38}$$

In particular, $f_{2,2}(\mu,\nu) < \infty$.

Proof. For any given pair of probability measures μ and ν in $\mathcal{P}(G_N^d)$, with centers $\boldsymbol{m}_\mu = \boldsymbol{m}_\nu$, we have

$$i\boldsymbol{k} \sum_{\boldsymbol{x} \in G_N^d} \boldsymbol{x} \mu_{\boldsymbol{x}} = i\boldsymbol{k} \sum_{\boldsymbol{y} \in G_N^d} \boldsymbol{y} \nu_{\boldsymbol{y}}.$$

For any transport plan π between μ and ν, we can rewrite the previous relations in the form

$$i\boldsymbol{k} \sum_{\boldsymbol{x},\boldsymbol{y} \in G_N^d} (\boldsymbol{x} - \boldsymbol{y}) \pi_{\boldsymbol{x},\boldsymbol{y}} = 0. \tag{2.4.39}$$

Using identity (2.4.39) we obtain

$$\begin{aligned}\hat{\mu}(\boldsymbol{k})-\hat{\nu}(\boldsymbol{k}) &= \sum_{\boldsymbol{x}\in G_N^d}\mu_{\boldsymbol{x}}e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}-\sum_{\boldsymbol{y}\in G_N^d}\nu_{\boldsymbol{y}}e^{-i\boldsymbol{k}\cdot\boldsymbol{y}}\\ &= \sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}\Big(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}-e^{-i\boldsymbol{k}\cdot\boldsymbol{y}}-i\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{y})\Big)\pi_{\boldsymbol{x},\boldsymbol{y}}\\ &= \sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}e^{-i\boldsymbol{k}\cdot\boldsymbol{y}}\big(e^{-i\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{y})}-1-i\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{y})\big)\pi_{\boldsymbol{x},\boldsymbol{y}}\\ &\quad+\sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}i\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{y})(e^{-i\boldsymbol{k}\cdot\boldsymbol{y}}-1)\pi_{\boldsymbol{x},\boldsymbol{y}}.\end{aligned}$$

Using that for all $\theta\in\mathbb{R}$

$$\begin{aligned}|e^{i\theta}-1|&\le|\theta|,\\ |e^{i\theta}-1-i\theta|&\le\frac{\theta^2}{2}\end{aligned}$$

we obtain

$$\begin{aligned}|\hat{\mu}(\boldsymbol{k})-\hat{\nu}(\boldsymbol{k})| &\le \frac{|\boldsymbol{k}|^2}{2}\sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}|\boldsymbol{x}-\boldsymbol{y}|^2\pi_{\boldsymbol{x},\boldsymbol{y}}+|\boldsymbol{k}|^2\sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}|\boldsymbol{x}-\boldsymbol{y}||\boldsymbol{y}|\pi_{\boldsymbol{x},\boldsymbol{y}}\\ &\le \frac{|\boldsymbol{k}|^2}{2}\sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}|\boldsymbol{x}-\boldsymbol{y}|^2\pi_{\boldsymbol{x},\boldsymbol{y}}\\ &\quad+|\boldsymbol{k}|^2\Big(\sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}|\boldsymbol{y}|^2\pi_{\boldsymbol{x},\boldsymbol{y}}\Big)^{\frac12}\Big(\sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d}|\boldsymbol{x}-\boldsymbol{y}|^2\pi_{\boldsymbol{x},\boldsymbol{y}}\Big)^{\frac12}.\end{aligned}$$

In particular, if we take π as the optimal transportation plan between μ and ν for the cost $|\boldsymbol{x}-\boldsymbol{y}|^2$ we get

$$\begin{aligned}\frac{|\hat{\mu}(\boldsymbol{k})-\hat{\nu}(\boldsymbol{k})|}{|\boldsymbol{k}|^2} &\le \frac{W_2^2(\mu,\nu)}{2}+\Big(\sum_{\boldsymbol{y}\in G_N^d}|\boldsymbol{y}|^2\nu_{\boldsymbol{y}}\Big)^{\frac12}W_2(\mu,\nu)\\ &= W_2(\mu,\nu)\left(\frac{W_2(\mu,\nu)}{2}+\left(\sum_{\boldsymbol{y}\in G_N^d}|\boldsymbol{y}|^2\nu_{\boldsymbol{y}}\right)^{\frac12}\right).\end{aligned}$$

Since

$$W_2(\mu,\nu)\le W_2(\mu,\delta_0)+W_2(\delta_0,\nu)\le\left(\sum_{\boldsymbol{x}\in G_N^d}|\boldsymbol{x}|^2\mu_{\boldsymbol{x}}\right)^{\frac12}+\left(\sum_{\boldsymbol{y}\in G_N^d}|\boldsymbol{y}|^2\nu_{\boldsymbol{y}}\right)^{\frac12},$$

and, as μ and ν are supported in $[0,1]^2$,

$$\sqrt{\sum_{\boldsymbol{x}\in G_N^d} |\boldsymbol{x}|^2 \mu_{\boldsymbol{x}}} \leq \sqrt{2}, \quad \sqrt{\sum_{\boldsymbol{y}\in G_N^d} |\boldsymbol{y}|^2 \nu_{\boldsymbol{y}}} \leq \sqrt{2},$$

and finally, we obtain (2.4.38):

$$\frac{|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|}{|\boldsymbol{k}|^2} \leq 2\sqrt{2} W_2(\mu,\nu).$$

□

We conclude by showing the validity of a reverse inequality, thus proving the equivalence between $f_{2,2}$ and W_2.

Theorem 9. *For any pair of measures $\mu, \nu \in \mathcal{P}(G_N^d)$, we have the inequality*

$$W_2^2(\mu,\nu) \leq \frac{T^3}{\pi} f_{2,2}(\mu,\nu).$$

Proof. Let π be the optimal transportation plan between μ and ν for the cost $|\boldsymbol{x}-\boldsymbol{y}|$, since $|\boldsymbol{x}-\boldsymbol{y}| \leq \sqrt{2}$ for all $\boldsymbol{x},\boldsymbol{y} \in G_N^d \subset [0,1]^2$, it holds

$$W_2^2(\mu,\nu) \leq \sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d} |\boldsymbol{x}-\boldsymbol{y}|^2 \pi_{\boldsymbol{x},\boldsymbol{y}} \leq \sum_{\boldsymbol{x},\boldsymbol{y}\in G_N^d} \sqrt{2}|\boldsymbol{x}-\boldsymbol{y}| \pi_{\boldsymbol{x},\boldsymbol{y}} = \sqrt{2} W_1(\mu,\nu).$$

Then, by Theorem 7 and Proposition 3 with $t=1$ and $p=s=2$, we get

$$\sqrt{2} W_1(\mu,\nu) \leq \frac{\sqrt{2}T^2}{2\pi} f_{1,2}(\mu,\nu) \leq \frac{T^3}{\pi} f_{2,2}(\mu,\nu),$$

which, together with the last inequality, concludes the proof. □

The previous bounds hold provided that μ and ν are centred in the same point. However, when $\boldsymbol{m}_\mu - \boldsymbol{m}_\nu \neq 0$, we can resort, as in Section 2.3, to the new metric

$$\mathcal{F}_{2,2}(\mu,\nu) := \sqrt{\left(f_{2,2}(\mu,\nu_{\boldsymbol{m}_\mu - \boldsymbol{m}_\nu})^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2\right)},$$

which is well-defined also for probability measures having different centres. This shows that we can generalize, similarly to Theorem 4 and Theorem 5, the equivalence of $\mathcal{F}_{2,2}$ and W_2 to measures which are not centred in the same point.

Remark 3. *Let us go back to what was said in Remark 1. From Equation* (2.4.34) *and Equation* (2.4.38) *we see that between W_1 and $f_{1,2}$ and W_2 and $f_{2,2}$ there are relations in which there is no numerical constant depending on the measure of the support space. These relationships tell us that these metrics are also able to take into account the geometric aspects of the measures under consideration. In Figure 2.3, we take as reference distribution a Dirac Delta centered in 0, δ_0, and we study how Total Variation distance (TV), $f_{1,2}$ ($\mathbb{F}$) and W_1 vary by taking as comparison measure a Dirac Delta centred in the integers*

from -20 to 20. As we can see, the TV is equal to 1 except when the comparison measure is equal to δ_0. Conversely, $f_{1,2}$ and W_1 have a V shape that mimics the fact that the comparison measure is approaching or leaving the reference measure in a geometrical/spatial sense.

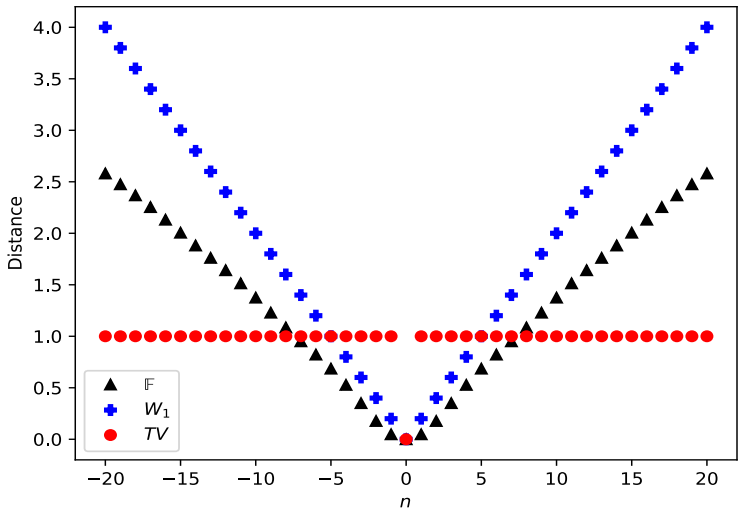

FIGURE 2.3: Behavior of W_1, $\mathbb{F}$ and TV metrics when comparing Dirac delta distributions. Results are re-scaled for visual convenience.

2.4.3 Connections with other distances

As discussed in [118], the case in which $s \leq 0$ leads to stronger metrics. In this case, we lose relations like (2.4.38), that link from above the Wasserstein metric with the Fourier-based metric. An interesting case is furnished by choosing $s = 0$ into (2.4.29). The metric, in this case, is defined by

$$\begin{aligned} f_{0,2}(\mu,\nu) &= \left(\frac{1}{|T|^d}\int_{[0,T]^d} |\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^2 d\boldsymbol{k}\right)^{\frac{1}{2}} \\ &= \left(\sum_{\boldsymbol{x}\in G} |\mu(\boldsymbol{x}) - \nu(\boldsymbol{x})|^2\right)^{\frac{1}{2}}, \end{aligned}$$

which defines the Total Variation distance between the probability measures μ and ν.

We remark that the distance above corresponds to the choice $\alpha = 0$, which does not require the measures to possess the same mass. In alternative one can choose a value $\alpha \in [0, 2)$. However, if $\alpha > 0$, one obtains a distance between measures that requires that the two measures have the same mass. Note however that the choice of values of $\alpha > 0$ allows obtaining a sequence of metrics that interpolate between the Total Variation distance and the W_1 distance, namely a family of measures that move from a strong metric to a weaker one.

In the case $s < 0$ the Fourier-based metric (2.4.29) becomes

$$f_{s,2}(\mu,\nu) = \left(\frac{1}{|T|^d}\int_{[0,T]^d} |\boldsymbol{k}|^{2|s|}|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^2 d\boldsymbol{k}\right)^{\frac{1}{2}}.$$

In particular, when $-s = n \in \mathbb{N}_+$, we find that

$$f_{-n,2}(\mu,\nu) = \left(\frac{1}{|T|^d}\int_{[0,T]^d} |\boldsymbol{k}|^{2n}|\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^2 d\boldsymbol{k}\right)^{\frac{1}{2}}.$$

This metric, by Fourier identity, controls the n-th derivative of the measures μ and ν.

2.5 Discretization of the Periodic Fourier-based metrics

When it comes to applications such as Image or signal processing, to compute the Fourier Transform, we rely on the Discrete Fourier Transform [119, 120]. In what follows, to simplify the notation, we consider one-dimensional discrete probability measures.

Definition 11 (Discrete Fourier Transform)**.** *The Discrete Fourier Transform of $\mu \in \mathcal{P}(G_N)$ is the N-dimensional vector $\hat{\mu} := (\hat{\mu}_0, \dots, \hat{\mu}_{N-1})$ defined as*

$$\hat{\mu}_k := \sum_{j=0}^{N-1} \mu_j e^{-2\pi i \frac{j}{N}k}, \qquad k \in \{0, \dots, N-1\}. \tag{2.5.40}$$

The Discrete Fourier Transform of a discrete measure can be expressed as a linear map:

$$(\hat{\mu}_0, \dots, \hat{\mu}_{N-1}) = \Omega \cdot (\mu_0, \dots, \mu_{N-1}), \tag{2.5.41}$$

where Ω is the $N \times N$ matrix defined as

$$\Omega := \begin{bmatrix} \omega_{0,0} & \omega_{0,1} & \dots & \omega_{0,N-1} \\ \omega_{1,0} & \omega_{1,1} & \dots & \omega_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{N-1,0} & \omega_{N-1,1} & \dots & \omega_{N-1,N-1} \end{bmatrix}, \tag{2.5.42}$$

and $\omega_{k,j} := e^{-2\pi i \frac{j}{N}k}$. Since the matrix Ω is invertible, the Discrete Fourier Transform is a bijective function.

Remark 4. *Since the complex exponential function $k \to e^{-2\pi i \frac{j}{N}k}$ is an $N-$periodic function for any integer j, we set $\hat{\mu}_k := \hat{\mu}_{mod_N(k)}$ for any $k \in \mathbb{Z}$, where $mod_N(k)$ is the $N-$modulo operation. In particular, $\hat{\mu}_{-k} = \hat{\mu}_{N-k}$ for any $k \in \{0, \dots, N-1\}$. In order not to burden the notation by dividing the cases with N even and N odd each time, we choose to fix N even from now on. For the N odd, all the results*

are valid, as long as you arrange the indices following what has been said in this remark

Remark 5. *If we consider the case of $\mu \in \mathcal{P}(G_N)$ with $G_N \subset [0,1)^2$, and if we denote $\mu_{j,l}$ as the (j,l) entry of the 2-dimensional tensor representing μ,* (2.5.40) *becomes:*

$$\begin{aligned}\hat{\mu}_{k,r} &:= \sum_{l=0}^{N-1}\sum_{j=0}^{N-1} \mu_{j,l} e^{-2\pi i \frac{j}{N}k + -2\pi i \frac{l}{N}r} \\ &= \sum_{l=0}^{N-1}\left\{\sum_{j=0}^{N-1} \mu_{j,l} e^{-2\pi i \frac{j}{N}k}\right\} e^{2\pi i \frac{l}{N}r} \qquad (k,r) \in \{0,\dots,N-1\}^2.\end{aligned}$$

This means that all the results and definitions for the one-dimensional case can be extended to the 2 or d-dimensional case considering each dimension separately.

We can now define the discrete counterparts of Periodic Fourier-based metric

Definition 12 (The Discrete Periodic Fourier-based Metric)**.** *Given μ and ν be two 1-dimensional probability measures over G_N, we define as*

$$Df^{(\alpha)}_{s,p}(\mu,\nu) := \frac{2}{|N|}\sum_{k=1}^{\frac{N}{2}} \frac{|\hat{\mu}_k - \hat{\nu}_k|^2}{|k|^{sp+\alpha}} \tag{2.5.43}$$

the (s,p,α)-Discrete Periodic Fourier-based Metric between μ and ν, where $p,s,\alpha \in \mathbb{R}$ and $\frac{N}{2}$ is the period of $\hat{\mu}$ and $\hat{\nu}$.

Recalling (2.4.32) and (2.4.37) we have

$$Df^{0}_{1,2}(\mu,\nu) := \frac{2}{|N|}\sum_{k=1}^{\frac{N}{2}} \frac{|\hat{\mu}_k - \hat{\nu}_k|^2}{|k|^{2}} \tag{2.5.44}$$

and

$$Df^{0}_{2,2}(\mu,\nu) := \frac{2}{|N|}\sum_{k=1}^{\frac{N}{2}} \frac{|\hat{\mu}_k - \hat{\nu}_k|^2}{|k|^{4}} \tag{2.5.45}$$

Finally, using (2.5.45), we define

$$D\mathcal{F}_{2,2}(\mu,\nu) := \sqrt{\left(Df_{2,2}(\mu,\nu_{\boldsymbol{m}_\mu - \boldsymbol{m}_\nu})^2 + |\boldsymbol{m}_\mu - \boldsymbol{m}_\nu|^2\right)}, \tag{2.5.46}$$

where in this case $\boldsymbol{m}_\mu$ is calculated as

$$\boldsymbol{m}_\mu = \sum_{k=0}^{N-1} k\mu_k.$$

Using the matrix representation of the Discrete Fourier Transform of (2.5.42), we can cast our Discrete Periodic Fourier-based metric in a matrix representation.

To this extent we introduce the $N \times N$ matrix $\mathbb{K}^{\alpha}_{s,p} := \text{diag}(k_z^{sp+\alpha})$, where

$$k_z := \begin{cases} 1 & z = 0 \\ \frac{1}{|z|} & z = \{1, 2, \dots, \frac{N}{2}\} \\ \frac{1}{\frac{N}{2} - (N-1-z)} & z = \{\frac{N}{2} + 1, \dots, N-1\} \end{cases}$$

Now, considering Remark 4, (2.5.43) read as

$$Df^{(\alpha)}_{s,p}(\mu, \nu) := \frac{1}{N}(\hat{\mu} - \hat{\nu})^T \mathbb{K}^{\alpha}_{s,p}(\hat{\mu} - \hat{\nu}) = \frac{1}{N}(\mu - \nu)^T \Omega^T \mathbb{K}^{\alpha}_{s,p} \Omega(\mu - \nu) \quad (2.5.47)$$

In this expression, we have incorporated the elements $\hat{\mu}_0$ and $\hat{\nu}_0$. However, $\mu \in \mathcal{P}(G_N)$, and hence:

$$\hat{\mu}_0 = \sum_{j=0}^{N-1} \mu_j e^{-2\pi i \frac{j}{N} 0} = \sum_{j=0}^{N-1} \mu_j = 1 \quad (2.5.48)$$

Since the same argument holds for ν, we have $|\hat{\mu}_0 - \hat{\nu}_0| = 0$.

Note that $\mathbb{H}^{\alpha}_{s,p} := \Omega^T \mathbb{K}^{\alpha}_{s,p} \Omega$ is a symmetric and circulant matrix, since $(\mathbb{H}^{\alpha}_{s,p})_{i,j} = \text{Re}((\hat{k}_z^{sp+\alpha})_{i-j})$. Therefore, its eigenvalues can be explicitly computed [121], leading us to the following result.

Lemma 4. *For any $p \geq 1$, the matrix $\mathbb{H}_p$ is positive definite and its eigenvalues are given by*

$$\lambda_i = N \cdot (k_z^{sp+\alpha})_i, \qquad i = 0, \dots, N-1.$$

Now recalling that every symmetric and positive definite matrix induce a norm [122], we can state that

Corollary 2. *Given $\mu,\ \nu \in \mathbb{R}^N$ two N-dimensional vectors. Then*

$$Df^{(\alpha)}_{s,p}(\mu, \nu) := \frac{1}{N}(\hat{\mu} - \hat{\nu})^T \mathbb{K}^{\alpha}_{s,p}(\hat{\mu} - \hat{\nu}) = \frac{1}{N}(\mu - \nu)^T \mathbb{H}^{\alpha}_{s,p}(\mu - \nu)$$

is a distance on $\mathbb{R}^N$. In particular, this also hold for all pair of vectors $\mu \in \mathcal{P}(G_N)$ and $\nu \in \mathcal{P}(G_N)$.

Remark 6. *Corollary 2 tells us that when we discretize the Fourier Transform, Equation (2.5.44) and Equation (2.5.45) define a distance on the whole $\mathbb{R}^N$ and not only on the restricted subset of vectors representing probability distribution. This is an interesting result since now we can apply these types of distances correlated to Optimal Transport out of the probability setting, reaching a broader horizon of applications.*

2.6 Numerical Results

In this section, we run three numerical experiments. The first two experiments are devoted to compare the Wasserstein metrics W_1 and W_2 with the corresponding Periodic Fourier-based metrics $f^0_{1,2}$ and $f^0_{2,2}$. We aim to highlight the

computational advantages in terms of the runtime of the $f^0_{1,2}$ and $f^0_{2,2}$ and show how these distances numerically relate to W_1 and W_2 respectively. The third experiment is conducted to show how $f^0_{1,2}$ can be applied to the general case of $\mathbb{R}^d$ vectors.

Implementation details. We implemented our algorithms in Python 3.7, using the Fast Fourier Transform implemented in the NumPy library [123]. To compute the Wasserstein distances, we use the Python Optimal Transport (POT) library [124] in the first experiment, while in the second one we use the Spatial-KWD library [125]. All the tests are executed on a MacBook Pro 13 equipped with a 2.5 GHz Intel Core i7 dual-core and 16 GB of Ram.

2.6.1 Experiment on DOTmark benchmark

The first experiment is devoted to extensively compare the numerical relations between Wasserstein metrics and Periodic Fourier-based metric. As problem instances, we use the DOTmark benchmark [117], which is a standard benchmark dataset for optimal transport problems in $2D$. The DOTmark benchmark contains 10 classes of grayscale images, each containing 10 different images. Every image is given in the data set at the following pixel resolutions: $32 \times 32, 64 \times 64, 128 \times 128, 256 \times 256$, and 512×512. Figure 2.4 shows the Classic, Microscopy, and Shapes images, respectively, (one class for each row), at the highest pixel resolution.

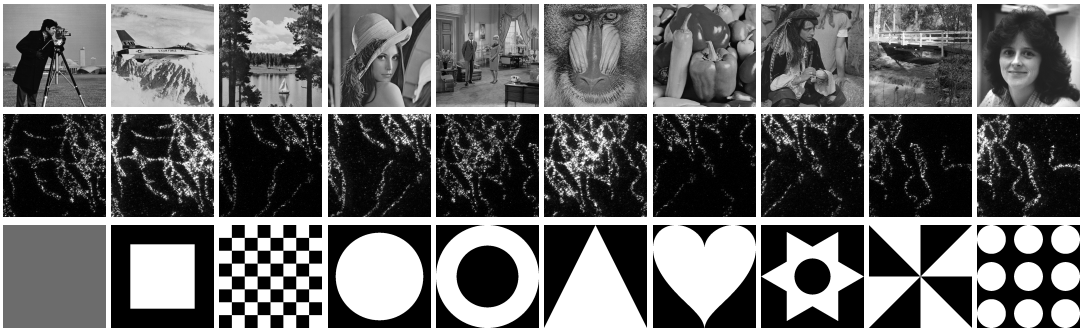

FIGURE 2.4: DOTmark benchmark: Classic, Microscopy, and Shapes images.

Results. For each pair of images of the DOTmark dataset, we have computed the reciprocal distance using the $W_1, W_2, f^0_{1,2}$ and $f^0_{2,2}$ distances, and the runtime in seconds.

The scatter plot in Figure 2.5 shows the relation between the W_2 and the $f^0_{2,2}$ distances for each pairs of images at pixel resolution 32×32. That plot shows that the two metrics are not only theoretically equivalent, as shown in Theorem 8 and Theorem 9, but also they return similar values in practice. The only exception is the Shape class, which, however, contains artificial shape images. On the more (application-wise) interesting Classic images, the two metrics return very close values.

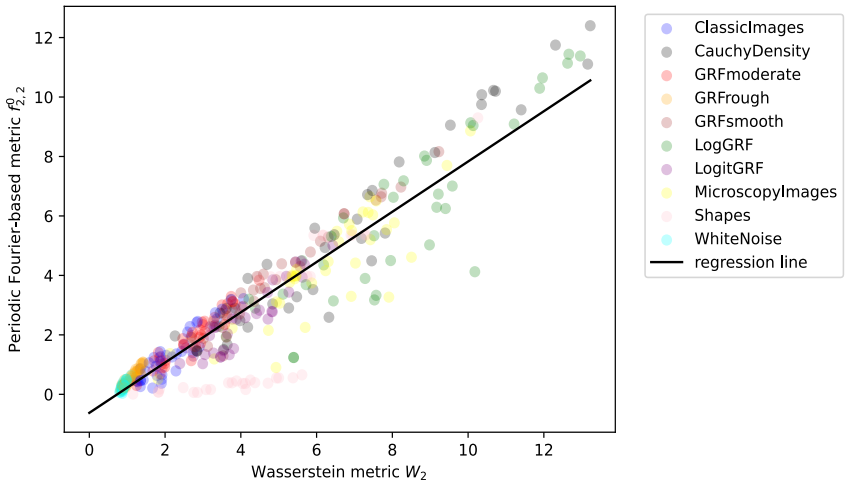

FIGURE 2.5: Wasserstein metric W_2 versus Periodic Fourier-based metric $f^0_{2,2}$: Comparison of distance values for 450 pairs of images of size 32×32.

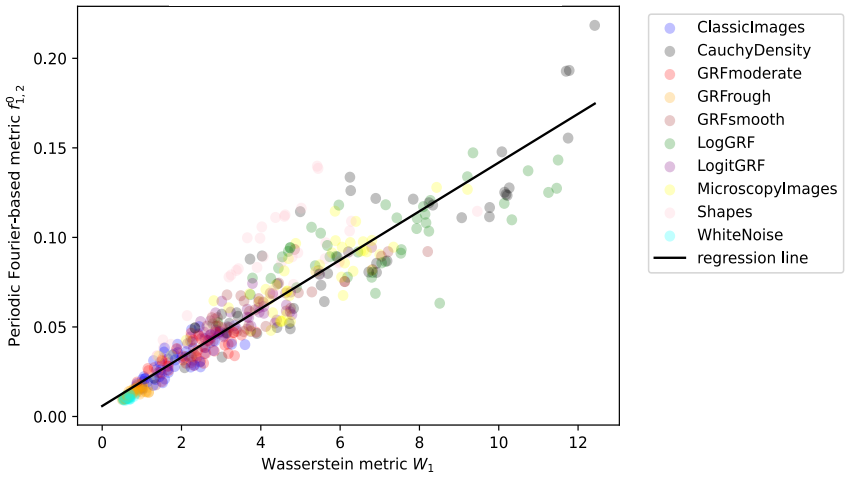

FIGURE 2.6: Wasserstein metric W_1 versus Periodic Fourier-based metric $f^0_{1,2}$: Comparison of distance values for 450 pairs of images of size 32×32.

Figure 2.6 reports the comparison between W_1 and the $f^0_{1,2}$. Even if in this case we do have not the good relation between the numerical value of the two computed distances as before, the two distances have a nice linear correlation. This suggests that the $f^0_{1,2}$ can be used as an alternative to the W_1 in classification or clustering tasks since it preserves similar mutual relations between samples. To highlight the linear dependence that exists in both cases we have reported the linear regression line.

Table 2.1 reports the averages and the standard deviations of the runtime, measured in seconds, at different image sizes. For each row and each metric, the averages are computed over 450 instances. The numerical results clearly show that the Periodic Fourier-based metric metrics are orders of magnitude faster, and permit evaluate the distance even for the largest 512×512 images in around

10 seconds. Note that using the POT library, we were unable to compute the W_1 and W_2 distances for images of size 256×256 and 512×512, due to memory issues.

TABLE 2.1: Runtime vs. Image size for different metrics: The runtime is measured in seconds and reported as "*Mean (Std-Dev)*". Each row gives the averages over 450 instances of pairwise distances.

	Averages Runtime in seconds			
Dimension	W_1	W_2	$f^0_{1,2}$	$f^0_{2,2}$
32×32	0.84 (0.30)	1.06 (0.32)	0.002 (10^{-4})	0.006 (10^{-4})
64×64	21.9 (7.96)	23.41 (8.49)	0.01 (10^{-3})	0.02 (10^{-3})
128×128	205.0 (45.9)	199.0 (45.0)	0.28 (0.07)	0.63 (0.16)
256×256			1.21 (0.40)	2.96 (0.94)
512×512			4.74 (1.32)	11.55 (2.84)

2.6.2 Comparison with KWD library

To confirm that the Fourier-based metric metrics well behave with respect to the Wasserstein metrics, we have decided to use another implementation of the Wasserstein distance and another dataset. The Spatial-Kantorovich Wasserstein Distance (Spatial-KWD) library [125] has been developed in C++ for [126] to compare true distributions and reconstructed distributions of density mobile phone maps starting from Mobile Network Operator (MNO) data. In this work,

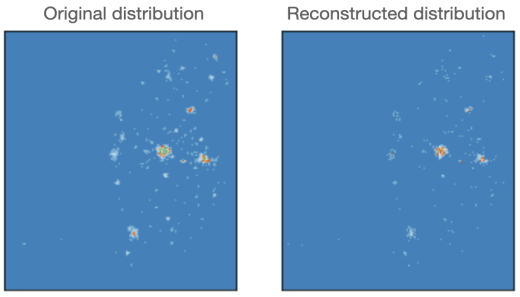

FIGURE 2.7: Original and Reconstructed spatial density of mobile phone distribution

we will not compare the various reconstruction methods, but we will compare the results obtained through the Spatial-KWD library and the Fourier-based metric $f^0_{1,2}$. To perform this comparison we have at our disposal 9 instances for each of the 4 different types of spatial resolution. The 4 spatial resolution are 267×228, 534×455, 1068×910, 2134×1819. Although 36 instances may seem few, they were carefully created to test the Spatial-KWD library during the software development phase. Testing on these instances therefore gives us reliable results on the behavior of the Fourier-based metric in relation to the Spatial-KWD library. As we can see from Figure 2.7, these densities

are very sparse. The Spatial-KWD library was optimized to gain advantages from the geometric configuration of these types of problems. For what concern the Fourier-based metric implementation, we do not perform any type of code optimization to leverage the sparsity and the geometry of the problem to speed up the calculation of the proposed Fourier-based metric.

As we can see from Figure 2.8, the correlation between the values calculated with the Spatial-KWD library and the values calculated with the Fourier-based metric $f^0_{1,2}$ are close to being linear in the two low resolution, while in the two high-resolution cases seem to be of quadratic type. This means that if we are using the Spatial-KWD library to estimate the goodness of fit between a real density and two different reconstructed densities, the two methods are equivalent as the order is maintained. However, we have a great advantage in terms of run time if we decide to use the Fourier-based metric. In fact for the instances with higher resolutions 2134×1819, the Fourier-based metric is calculated in around 17.5 seconds, while the runtime of the Spatial-KWD is in the range between 2746 and 33195 seconds, that is at least half an hour for a single comparison.

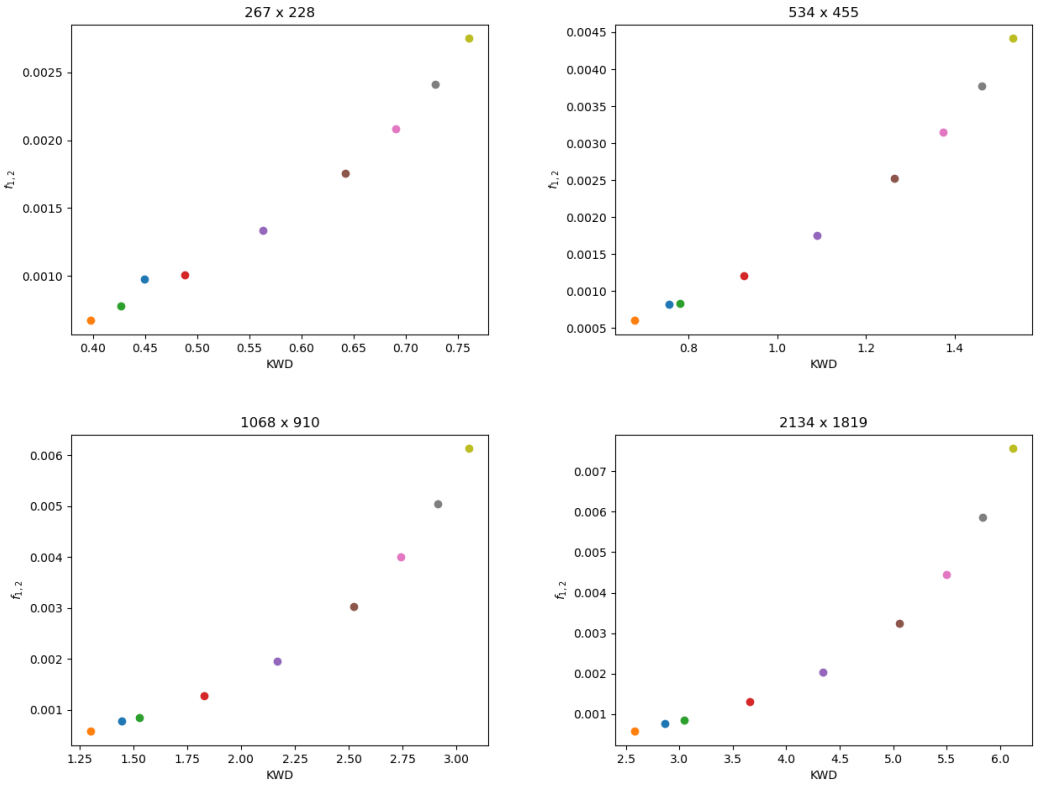

FIGURE 2.8: Correlation between Spatial-KWD and $f^0_{1,2}$.

2.6.3 Experiment on ECG5000

Since the Fourier-based metric can be extended also to vectors outside the probability simplex (see Remark 6), we have decided to evaluate the performance of the $f^0_{1,2}$ metric as a classification tool. Since the Fourier transform is widely used in signal processing, we have implemented a k-Nearest Neighbor (k-NN)

TABLE 2.2: Comparison between $f^0_{1,2}$ and L_2 distance on anomaly detection with different values of k. F1 score and Accuracy are reported in percentage.

metric	k=1 F1/Acc	k=3 F1/Acc	k=5 F1/Acc	k=7 F1/Acc	k=9 F1/Acc
$f^0_{1,2}$	0.959/0.966	0.963/0.970	0.965/0.972	0.967/0.973	0.968/0.974
L_2	0.793/0.835	0.853/0.887	0.877/0.905	0.887/0.913	0.891/0.916

classifier to detect anomalies on the ECG5000 dataset [127]. This dataset is composed of 5000 earth beats (500/4500 train/test split) labelled in five classes: Normal (N), R-on-T Premature Ventricular Contraction (R-on-T PVC), Premature Ventricular Contraction (PVC), Supra-ventricular Premature or Ectopic Beat (SP or EB) and Unclassified Beat (UB). The dataset is highly unbalanced. The prevalent class is the normal one with 292/2627 samples in the train/test split, followed by Ron-T PVC with 177/1590 samples, then SP with 19/175 and finally PVC with 10/86 samples. Since our goal is to classify normal vs anomaly heartbeats, we have dropped the UB class (2/22 samples), since we do not know the nature of the samples in this class. To classify each beat in the test set we evaluate the distance between the beat and all the beats in the training set. Then we have assigned the most present label among the k-elements with the smallest distance from the sample in question. We selected as k the values 1,3,5,7,9 and run the same experiment comparing $f^0_{1,2}$ with the standard L_2 distance.

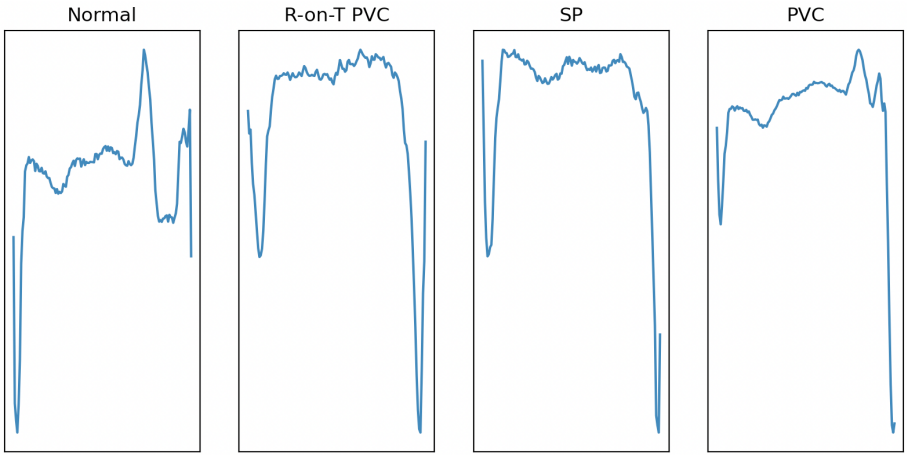

FIGURE 2.9: Samples of the four classes in the ECG5000 datset.

Results. In Table 2.2 we have reported F1 score and Accuracy as k varies:

$$\textbf{F1 score} \qquad F1 := \frac{TP}{2TP + FN + FP},$$

$$\textbf{Accuracy} \qquad Acc := \frac{TP + TN}{FN + FP + TP + TN},$$

where TP, TN, FP, FN are computed on the change class and represent the true positives, true negatives, false positives, and false negatives respectively.

From these results, it is straightforward to see that $f^0_{1,2}$ over performs the L_2 distance. This is an interesting result if we recall that, thanks to the Parseval theorem, we have

$$\int_{[0,T]^d} |\mu(\boldsymbol{x}) - \nu(\boldsymbol{x})|^2 d\boldsymbol{x} \propto \int_{[0,T]^d} |\hat{\mu}(\boldsymbol{k}) - \hat{\nu}(\boldsymbol{k})|^2 d\boldsymbol{k}.$$

This means that weighting the frequency is not only crucial in the abstract setting to gain equivalence with the W_1, but also has an important role in the application scenario. In this case, we have weighted the frequency according to the (2.4.29). Since in this case the two vectors representing the two ECGs are not probability measures, we have assigned a weight of 1 to the first entry of the Fourier Transformed vector.

We also compared the best results obtained with k=9 with results reported in [128]. In [128–130] the authors have trained a Neural Network composed with Long Short-Term Memory (LSTM) [131] blocks and convolution to classify time series. In particular, in [128] the authors propose both a supervised and unsupervised model, coupling in the first case, the LSTM network with a Support Vector Machine (SVM) [132] and in the second case with a clustering algorithm. An architecture that incorporates both LSTM and convolution blocks is proposed in [129], but only in the supervised scenario. Also in [130] the LSTM blocks are used, but in this case in an Encoder-Decoder model to perform anomaly detection using reconstruction error as anomaly score. A Dynamic Time Warping (DWT) [133] distance clustering method is presented in [134].

TABLE 2.3: Comparison between $f^0_{1,2}$ k-NN anomaly detector with the results reported in [128].

Model	Acc	F1
VRAE+SVM [128]	0.984	0.984
VRAE+Clust/W [128]	0.959	0.952
F-t ALSTM-FCN [129]	0.949	-
SAE-C [130]	0.934	-
oFCMdd [134]	-	0.808
$f^0_{1,2}$ k-NN	0.974	0.968

As we can see from Table 2.3, our k-NN approach attains competitive results with other state-of-the-art methods in the two class settings (Normal vs Anomaly). Comparing our results with [134], another method explicitly using a distance to directly compare time series, we can conclude that the $f^0_{1,2}$ could be an interesting metric alternative to the well-known and used DWT. With respect to [128–130], our anomaly detector has comparable performance, but does not require any training phase. This makes our pure metric approach a suitable

alternative to the learning-based approach in the context of a scarce number of samples or imbalanced datasets, which is a common scenario in industrial or healthcare scenarios.

Chapter 3

TinyCD

In this chapter, we introduce the Line Clearance problem, and we introduce TinyCD, a (not so) deep learning model, developed to deal with this problem in an industrial scenario. As the name suggests, Line Clearance consists in monitoring the cleanliness of a production line, in particular during the packaging and quality control of drugs in the pharmaceutical sector. Typically, the production line is composed of different machines, each of which has a specific task. Some of them are: positioning the tablets inside the blisters, labelling vials, and checking that the product belongs to the correct batch. Especially in the pharmaceutical field, the packaging process must follow strict protocols to ensure the integrity of the products. It is therefore necessary that all the machines are checked at the very end of a production lot, and at the very beginning of the next one; this is needed to prevent, for example, the mixing of different products. In addition, it is possible that a dangerous object, due to machinery vibrations or other accidental causes, gets stuck or is mistakenly left on a machine after maintenance. In these cases, starting the machinery could lead to a breakdown. To avoid this, the cleanliness and integrity of the machinery are checked regularly by specialized operators. These checks require the production line to be stopped until the operator has completed the inspection. Given the size and complexity of these machines, the inspection phase can take several tens of minutes, during which the production is stopped, resulting in a loss of efficiency.

To overcome most of these drawbacks, a computer vision system could be adopted. The use of micro-cameras on the machines, coupled with a segmentation or object detection model, can monitor the crucial parts of the machinery in dramatically reducing the inspection times, thus saving production efficiency. To solve this task, several challenges must be faced, most of which are related to the specific nature of the problem. In addition to the classic environmental variables such as variations of lighting conditions and the presence of casted shadows, in this scenario, our model must be able to recognize potentially harmful objects without knowing them a priori. As an example, it is possible that a pad or screw gets stuck on the production line, or that a screwdriver is inadvertently left on a conveyor belt. Moreover, considering that the machines can be scpecifically configured according to the product they must work on, checking that the configuration is the correct one allows to reduce the erroneous production of compromised batches. Finally, notice that production machines have moving parts, thus on an opposite side the model must be robust to admissible changes related to them.

In the world of machine-learning, models are trained to accomplish tasks

by means of training examples, generally collected in datasets. Collecting a dataset containing all the needed cases for each machine is, at least from a business point of view, a solution to be avoided as it is very expensive in terms of time and resources. A standard should also be defined for categorizing dangerous objects and under what conditions they must be considered dangerous. Moreover, the model must work in (almost) real-time on devices with limited computational capacity, eventually processing dozens of image streams acquired by many cameras from the production line.

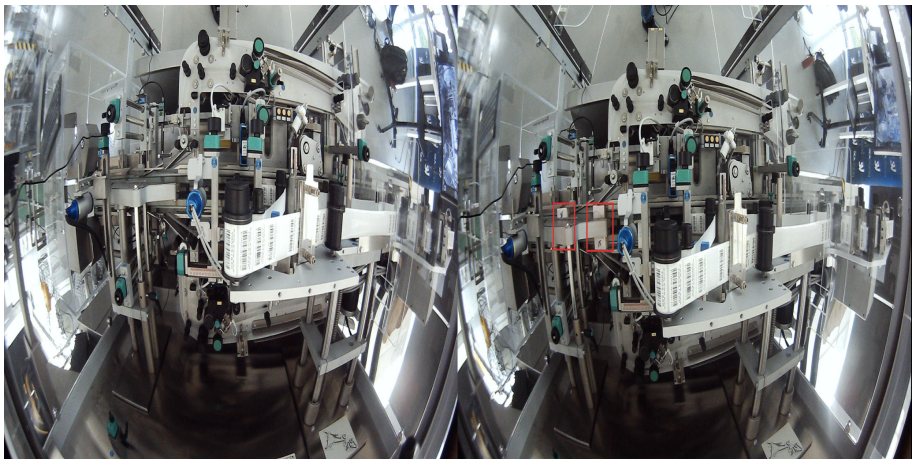

FIGURE 3.1: Example of a machine to be monitored. Left: clean machine. Right: machine with two cases interlocked in line. We have highlighted the cases with two red bounding boxes. Images are juxtapposed to highlight the spatial shift.

To takle this problem, we decided to adopt the tools offered by the Change Detection field. The peculiarity of this kind of models is that they are able to highlight changes between two inputs. To teach the model which changes are important and which should be ignored, during the training phase, pairs of images with and without the changes are shown to the model under different conditions. It will therefore be sufficient to couple images with different conditions, considered acceptable changes, to force the model to not track those types of variations. From an industry point of view, directly comparing two images also has another advantage: it is possible to decide from time to time what is the normal state of the machine, without the need to re-train the model.

Change Detection is one of the main research topics in the Remote Sensing community. Also, in applications of the aerial imaging, the Change Detection models can compare two co-registered images I_1 and I_2 acquired at times t_1 and t_2 [46, 135, 136]. In this scenario, some relevant changes are: urban expansion, deforestation, or post-disaster damage assessment [137–143]. On the other side, some irrelevant changes are: lighting conditions, shadows, and seasonal variations. Since a lot of research work has been devoted by the Remote Sensing community to this field, we decided to start from the existing models to validate the approach also on our industrial project. In our work, not only we faced the problem from a qualitative point of view, but we taked care also of the computational complexity exposed by the designed model. Notice that, as reported in [144], the state-of-the-art models count several million parameters,

and at least 4 GigaFlops for the processing of a single image, which badly fits our production requirements.

To present our ideas, and to fairly test our model by comparing it to the other state-of-the-art ones, we compared our results in the context of Change Detection on aerial images. To this extent, in Section 3.1 we introduce in detail the Change Detection problem in the aerial images' context, and we discuss the relevant literature in Section 3.2. In Section 3.4 we present our model and in Section 3.5 we show the obtained results with a detailed comparison with other state-of-the-art models and the ablation study on our model. Finally, in Section 3.6, we discuss the obtained results, and in Section 3.7 we present preliminary results in the industrial scenario.

3.1 Change Detection on Aerial Images

Thanks to the increasing number of available high resolution aerial imaging datasets, such as [137, 141, 142], data driven methods like deep Convolutional Neural Networks found successful applicability [145]. The well known ability of deep Convolutional Neural Networks to extract complex and relevant features from images is the key factor for their early promising results [146]. In the Change Detection scenario, complex features are not sufficient to accomplish the task. To detect the occurred changes, it is in fact crucial to model the spatio-temporal dependencies between the two images. Unfortunately, plain Convolutional Neural Networks have a limited receptive field due to the usage of fixed kernels in convolutions. To overcome this issue, recent works focused their attention to enlarging the receptive fields by employing different kernel types [147], or by adding attention mechanisms [137, 141, 148, 149, 149–151]. However, most of them failed to explicitly relate data in the temporal domain, since attention mechanisms are applied separately on the two images. The self attention mechanism adopted in [137, 151] shows promising results relating images in the spatio-temporal domain. More recently, Transformers have been introduced in Change Detection because of their receptive fields spatially covering the whole image [144, 152]. Notice that, by applying multi-headed attention layers in the decoder part of the network, the receptive field covers the temporal domain too. Unfortunately, the resulting models are computationally very inefficient.

The Change Detection field finds applicability also outside the remote sensing world. As an example, in [143, 153] two models are discussed in order to be used on drones or other autonomous vehicles to implement smart city monitoring functions. In our case, the change detection model has been developed for an industrial application. In our application field, the need for real-time performances adds a model complexity constraint. Unfortunately, the majority of state-of-the-art models are millions-parameters-sized, so that their applicability is not possible. Another issue with those big models is related to the training time clearly affected by the size of the model. With large models, the Hyper-Parameters-Optimization task requires resources that are usually not available to medium-small companies. Moreover, big networks require dedicated hardware also at inference time. This is in contrast with production requirements and

project budgets. The search for models having both small size and performances comparable to the current state-of-the-art can be considered an open problem.

A possible strategy to cope with both model size and complexity that, to the best of our knowledge, has not been studied in the literature, is to use low-level features to compare the two images under examination. Another underestimated aspect, in our opinion, is that a Siamese type backbone produces two tensors containing channels arranged semantically in the same order. This observation could be used to design strategies for merging features more efficiently.

The main purpose of our work is to investigate the aforementioned issues, developing a neural network that requires lower computational complexity with respect to the state-of-the-art Change Detection models, reaching at the same time comparable performances.

The major contributions of our work are the following:

- We explore the effectiveness of using low-level features in the problem of comparing images. The results validate our intuition that in this context the low-level features are sufficiently expressive. Moreover, this allowed us to significantly limit the number of model parameters.

- We introduce a novel strategy to mix the features between the two images. This strategy allows the computation of a spatio-temporal correlation between the input images keeping a low computational complexity.

- A fast attention mechanism is introduced with a block called MAMB. It uses features localized in space to compute attention masks needed in the up-sampling phase to refine the low-resolution results.

- We propose to use a pixel-wise classifier to generate the final mask. In our tests, this proved to be very effective.

Our architecture exploits the information contained in the channels of the feature vectors generated by the backbone. For this reason, it can effectively exploit low level features such that a relatively small backbone can be adopted. Being the backbone the most time-consuming and parameter-demanding component in the architecture, especially in Siamese architectures where it is evaluated twice, maintaining it as small as possible allows us to achieve our goal. In particular, we are able to maintain the total number of parameters below 300000.

Finally, we compare the quality of the model with state-of-the-art architectures, and we demonstrate that it has performances comparable if not even superior to other state-of-the-art models in the Change Detection field. We have extensively tested our model on public and proprietary datasets. In this chapter we highlight the results obtained in the field of aerial images on public datasets.

3.2 Related works

3.2.1 Early deep neural network works on Change Detection

Deep Learning models, and in particular Convolutional Neural Networks, have been applied with great success in image comparison tasks [154–156], in pixel-level image classification [11, 157, 158], and they represent the state-of-the-art in many other Computer Vision fields [159].

Models in the context of the Change Detection must manage two inputs: one image I_1 acquired at time t_1, and another one I_2 acquired at time t_2. The correct use of these two inputs, and the features extracted from them, are extremely important for the well behavior of the Change Detection model. One of the first works that applies deep learning techniques to the field of Change Detection is [160]. This work highlights how deep neural networks, in particular Deep Belief Networks obtained by stratifying Restricted Boltzmann Machines, are a very effective tool to compare and highlight the changes between the two images under examination. To the best of our knowledge, the first work that applies Convolutional Neural Networks to the Change Detection problem is [146]. The authors propose two different approaches. In the first case they use a U-Net [11] type network with the Early Fusion Strategy (FC-EF), i.e. they concatenate the images I_1 and I_2, and then they feed the U-Net with the resulting tensor. In the second case, they investigate the Feature Fusion Strategy. To this aim, they employ a Siamese U-Net type network [155, 158, 161] where the two images are processed separately, and subsequently the features are fused in two different ways: concatenation (FC-Siam-conc) and subtraction (FC-Siam-diff). These fused features are then used as skip connections in the decoder. After this seminal work, an entire research line investigated both the Early Fusion Strategy [138, 149, 162, 163], and the Feature Fusion Strategy [137, 141, 147, 148, 150, 151, 164–169].

To take full advantage of the large amount of spatial information, deeper Convolutional Neural Networks such as ResNet [45] or VGG16 [170] have been used [137, 141, 147, 151] in order to extract spatial information and group them in a hierarchical way. Unfortunately, standard convolution has a fixed receptive field that limits the capacity of modelling the context of the image. To face this issue, atrous convolutions [171] have been experimented [147]: they are able to enlarge the receptive field of convolutional kernels without increasing the number of parameters.

3.2.2 Attention based Convolutional Neural Network

To definitively overcome the problem of fixed receptive field, attention mechanisms, in the forms of spatial attention [141, 148, 149], channel wise attention [141, 148–150], and also self-attention [137, 151], have been introduced. In [141], the attention mechanisms are used in the decoder part: the channel wise attention is used to re-weight each pixel after the fusion with the skip connections, while the spatial attention is adopted to spatially re-weight the

pixels containing misleading information due to the up sampling step. To further exploit the interconnection between spatial and channel information, in [148] a dual attention module has been introduced. The co-attention module introduced in [150] tries to leverage correlation between features extracted from both images. Also in [150] a co-layer aggregation and a pyramid structure is used to fully exploit the features extracted at each level and with different receptive fields. In [137], the non-local self attention introduced in [172], have been applied to Change Detection. This mechanism consists in stacking the features extracted from a Siamese backbone, and to apply both a basic spatial attention mechanism, and a pyramidal attention mechanism. Since these two attention blocks are applied to stacked features obtained from I_1 and I_2, these are correlated in a non-local spatio-temporal way. Another interesting approach is the one presented in [173]. In this paper, the authors decided to combine Convolutional Neural Networks with Object Image Analysis (OBIA) mitigating the limited receptive field problem. In a first preprocessing phase, they segment the image and extract the patches containing objects to be compared. Subsequently, the extracted patches are compared using a Convolutional Neural Network which then works on small patches containing more specific and detailed information.

In [153], the authors propose a temporal attention mechanism. They exploit the features extracted from I_2 to generate a query matrix which is then compared with the features extracted from I_1. This mechanism is made dynamic by reducing the receptive field as tensors' spatial dimension diminishes. Finally, the authors also use attention mechanisms capable of emphasizing some horizontal and vertical dependencies of recurring objects in their scenario.

Convolutional block attention modules (CBAMs), composed by a channel attention block and a spatial attention block, are staked and integrated in the features' extraction block of [174]. These blocks are connected with the residual outputs of every block of the ResNet18 based siamese backbone, in order to fully capture the effective information in multi-scale features. An interesting feature of [174] is the coupling of the Change Detection network with a GAN based super resolution module, thus extending the Change Detection applicability to images with different resolutions.

3.2.3 Transformers in Change Detection

The global attention mechanisms introduced with Transformers [175, 176] have also been applied to the Change Detection problem. In [144] the authors employ a modified ResNet18 as Siamese backbone to extract features. Then, to better justify the use of Transformer blocks, they follow a parallelism between the natural language processing field, and the image processing one, by introducing the semantic tokens. Roughly speaking, semantic tokens are the pixels of the last feature tensor extracted by the backbone. The authors use this concept to illustrate that concatenating single pixels and then processing them with a transformer encoder-decoder, a pair of feature-tensors can be obtained incorporating both global spatial information, and global temporal information. On the other side, in [152] the authors replace the Convolutional Neural Network backbone with a transformer in order to exploit the global information contained in the

images right from the start. In this model, the temporal aggregation is done only in the final multilayer perceptron decoder.

3.2.4 Relations between our work and existing models

Our work is inspired by [144]. As reported in Section 3.2.3, in that work the authors introduce the concept of semantic tokens, which are basically single pixels of the tensor obtained by the backbone. Then, they use a transformer in order to process these tokens and extract global spatio-temporal information. In agreement with [144], we believe that the information contained in the pixel/semantic token is crucial to obtain a good result. However, we prefer to apply channel-wise local feature comparison, limiting the semantic complexity and aggregation of adopted features to the first few backbone layers; whilst in [144] the comparison is global, being it obtained by means of transformers. Moreover, we adopt Multi Layer Perceptrons (MLPs) to compute both the spatial attention maps and the final mask, actually facing the problem as a pixel-wise classification one. Recently, MLP blocks have received great attention in computer vision community [177–183]. These architectures divide the images into patches and then process them with MLP blocks. Different structures of MLP blocks have been proposed to incorporate as much spatial information as possible. For example, in [178, 183] a spatial shift operator is applied in order to obtain information from different axial directions. The CycleMLP block proposed in [177] follows a similar idea but instead of applying the spatial shift operator to the features' tensor, it composes several MLP steps capable of mimic the shift. A more refined version of these concepts is proposed in [179] were the authors employ a block which dynamically learns the spatial offset used in CycleMLP. In our model, the MLP blocks work exclusively along the channel dimension to both compute the spatio-temporal attention maps, and produce the final pixel-wise classification.

3.3 Backgrounds on Convolution and Attention

In Section 3.2, reviewing the literature on Change Detection, we have repeatedly mentioned the Convolution operator and the Attention mechanism. In this section, we give an introduction to these two operations as they are two of the fundamental building blocks of deep learning models for computer vision. A more detailed discussion of these topics can be found in [42].

3.3.1 The Convolution operator

The notion of convolution sounds very familiar to those with a mathematical background. We start from the formal definition of the convolution. Indicating with $L^p(\mathbb{R}^N)$ the space of functions *modulo* p integrable in the sense of Lebesgue, that is $g \in L^p(\mathbb{R}^N) \Leftrightarrow \left(\int_{\mathbb{R}^N} |g(x)|^p dx\right)^{\frac{1}{p}} \leq +\infty$, we define

Definition 13. *Given $f \in L^1(\mathbb{R}^N)$ and $k \in L^p(\mathbb{R}^N)$, with $1 \leq p \leq +\infty$, we can define the function $f * k \in L^p(\mathbb{R}^N)$ as*

$$(f * k)(x) := \int_{\mathbb{R}^N} f(x - y)k(y)dy \tag{3.3.1}$$

To be more formal, Definition 13 should actually be stated and proved as a theorem. However, this is not the place to go into the details of Lebesgue's measure theory, and we refer the reader to [184] for a more precise and formal treatment of measure theory and integration.

What interests us in this discussion is understanding the meaning of (3.3.1). What (3.3.1) tells us is that the convolution operator $f * k$ is calculating a weighted average of the function f using the function k, which from now on we will call *kernel*, as the weight function. This weighted average operation has several applications in mathematics and is a very powerful tool. For example, in functional analysis, the convolution between L^p functions, which have no regularity from the point of view of the derivatives, with kernels with high regularity and compact support, called mollifiers, allows proving density results of spaces of regular functions in L^p spaces [184, Theorem IV.23].

To see how it works in the discrete context of image processing and to get an even clearer idea of how convolution works, let us use Figure 3.2. What is called Input in the Figure 3.2 is our f of (3.3.1) and can be interpreted as a single channel image, for example, a grayscale image. If we interpret the convolution as a dynamic process, we can think that the kernel k starts acting on the first 4 pixels of the image and through (3.3.1) returns the weighted average of those pixels. After that, in the following steps, the kernel moves on the image returning the weighted averages of the other parts of the image.

Let us immediately make a couple of observations regarding on how convolution works and is implemented. The attentive reader will have noticed that what is reported in Figure 3.2 is not the discretization of (3.3.1). In fact, in (3.3.1) the function f is flipped with respect to the kernel k in the argument y ($f(x-y)$ vs. $k(y)$). In other words, what is shown in the figure is the discretization of the

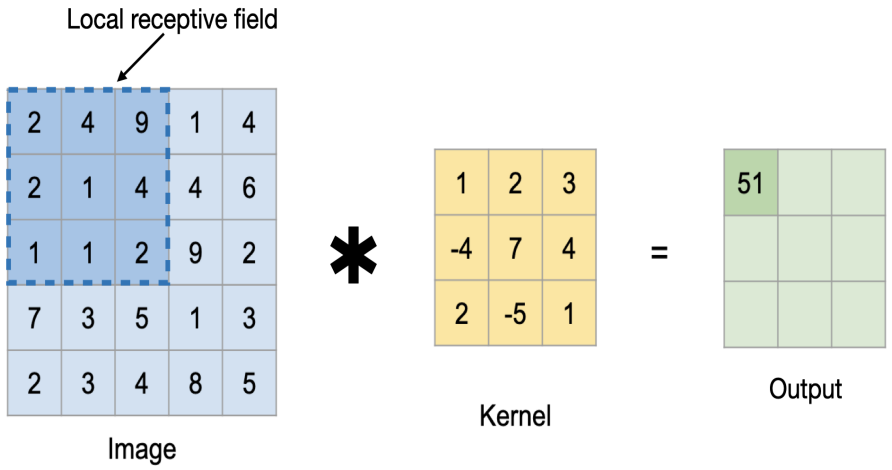

FIGURE 3.2: Visual explanation of convolution operator on 2 dimensional discrete domain.

cross correlation, which can be expressed as

$$\int_{\mathbb{R}^N} f(x+y)k(y)dy. \tag{3.3.2}$$

From a mathematical point of view, the main difference is that convolution satisfies commutativity, i.e. $f * g = g * f$, and cross correlation does not satisfy this property, which is a very useful property for example when writing proofs. Since, apart from this distinction on the sign, convolution and cross-correlation implement the same type of operation, and given that commutativity does not play a relevant role in computer vision, computer vision and machine learning libraries usually implement the operation of cross-correlation, calling it convolution by convention, probably to associate the operation at least intuitively with the better known mathematical convolution. The second thing we want to notice is the size difference between the input f and the result $f * k$. In the case shown in Figure 3.2, we start from a 5×5 input and obtain a 3×3 output. This is because the kernel, 3×3 in size, in this case, is scrolled over the image by superimposing it as shown without ever exceeding the edges of the image. In computer vision libraries, convolution is implemented by adding two parameters, called *padding* and *stride*, through which this behaviour can be changed. Padding consists of inserting values, usually 0, around or inside the image so that the final result has the same, or even greater, spatial dimensions as the input. Stride on the other hand controls how much the kernel shifts on the image and can be used instead to decrease the spatial dimensions. Combining kernel size, padding and stride we can get combinations suitable for every need. In the case of multi-channel images, for example RGB images, the convolution is calculated using kernels with a spatial dimension chosen by the user and many channels equal to that of the image, in this case, 3. The situation remains similar

when the input has C channels: we will use kernels with C channels. With this choice, the output has spatial dimensions according to the kernel spatial dimensions, padding and stride choices, but whose number of channels is 1. In other words, each kernel calculates the weighted average of the input not only spatially but also to all channels. For this reason, convolutional layers in neural networks implement in parallel the action of many kernels on the input equal to the number of desired channels on the output. In this way, different information obtained by making different weighted averages of the information contained therein is simultaneously extracted from an image or a tensor of features. For clarity, we underline that the strategy just described is only the simplest one and historically the first to be used. Over the last few years, new strategies for implementing convolutional layers have been explored. We will see for example in Section 3.4.3 the use of grouped convolutions.

So far, we have described the idea behind convolution and its implementation. What remains for us to understand is why convolutional networks have been so successful. We could think that the kernels for convolution operations can be implemented by an expert user. However, finding the kernel for each feature we want to extract can be a long and tedious job and automating this process is one of the reasons for the success of deep learning. Deep learning not only automates the process but can learn kernels that are effective for various tasks directly from data. This is made possible by the revolutionary design of convolutional neural networks, which mimic the way our own visual cortex processes information. Thanks to groundbreaking research on the brain of cats [41], scientists discovered that the brain breaks down visual information into layers, gradually building up complexity from simple lines and points to more advanced shapes and objects. Similarly, convolutional neural networks stack layers upon layers of convolutions, creating networks that can handle an extraordinary amount of information with ease. As stated in [42], convolutional neural networks can be considered as "the greatest success story of biologically inspired artificial intelligence."

Since the spatial dimension of the kernel affects the computational complexity, to maintain a good trade-off between performance and execution speed, we choose to use kernels with limited spatial dimensions. However, this choice means that the convolutional layers fail to model long-range dependencies within the image/tensor. Using the convolution operator we can reduce the spatial dimension of the image/tensor by concatenating several convolutional layers and hence aggregate information. Several variants of the classical convolution have also been developed, such as the atrous convolution [171] which consists in using a 0-padding inside the kernel by increasing its receptive field without increasing the computational complexity. However, even these solutions fail to extract global information. In the next section, we see the attention mechanism whose intent is to incorporate information from the whole image/tensor.

3.3.2 Attention mechanism

The attention mechanism was introduced in [185] in a neural encoder-decoder architecture for language translation. In the context of natural language processing (NLP), it is clear how important it is to have a mechanism capable of taking into account long-range dependencies within one or more words. The idea from which the attention mechanism was born is precisely that of developing a layer capable of focusing on the important regions within a context. In [176] Transformers architectures, composed of attention blocks interspersed with fully connected layers, were introduced. Transformers currently represent the state of the art in NLP, confirming the validity of the attention mechanism. Driven by the excellent results obtained in NLP, the researchers then exported the same ideas in other fields such as computer vision. To give a formal idea of the attention mechanism in computer vision, we follow [172, 186]. For a complete overview of all the attention mechanisms developed in computer vision, we refer the reader to [187].

Consider a tensor $X \in \mathbb{R}^{C\times H\times W}$, where C is the number of channels, H and W stand for height and width. We call pixel the element $x_{C,i,j}$, that is the vector at position i, j with length C. Where it is not necessary, we omit the reference to the channel dimension for the ease of notation. Let us fix a neighborhood $\mathcal{N}_d$ of pixel $x_{i,j}$ of height and width equal to d. The same dimension for height and width is set for convenience only. Now we can introduce the three fundamental elements of the attention layer:

Query $q_{i,j} := W_Q x_{i,j}$. This is the reference vector for which we want to calculate the output value according to the attention mechanism, processed through the W_Q matrix;

Keys $K \in \mathbb{R}^{C'\times d\times d}$ where each element $k_{r,s} := Q_K x_{r,s}$ for all $x_{r,s} \in \mathcal{N}_d$;

Values $V \in \mathbb{R}^{C'\times d\times d}$ where each element $v_{r,s} := Q_V x_{r,s}$ for all $x_{r,s} \in \mathcal{N}_d$.

The learnable weights of the attention layer are the three matrices W_Q,W_K,$W_V \in \mathbb{R}^{C'\times C}$, where C' is set by the user.

Then, the output vector $y_{i,j} \in \mathbb{R}^{C'}$ is obtained by

$$y_{i,j} = \sum_{(r,s)\in\mathcal{N}_d} \text{Softmax}_{(r,s)}(q_{i,j}^T k_{r,s}) v_{r,s}, \tag{3.3.3}$$

where, with a little abuse of notation, we have used $(r, s) \in \mathcal{N}_d$ to indicate that the operation is made for all the element that comes from the neighborhood $\mathcal{N}_d$. This process is repeated for all pairs of indices i,j, obtaining $Y \in \mathbb{R}^{C'\times H\times W}$. In words, for each vector of the tensor the corresponding processed vector called Query is calculated, and for each element of its neighborhood the matrices of Keys and Values are calculated. All these elements are then combined following (3.3.3). We can actually generalize (3.3.3) by writing

$$y_{i,j} = \sum_{(r,s)\in\mathcal{N}_d} f(x_{i,j}, x_{r,s}) g(x_{r,s}), \tag{3.3.4}$$

where $f : \mathbb{R}^C \times \mathbb{R}^C \to \mathbb{R}$ is a function that computes a scalar relationship between $x_{i,j}$ and $x_{r,s}$, and $g : \mathbb{R}^C \to \mathbb{R}^{C'}$. Figure 3.3 illustrates the attention mechanism described so far.

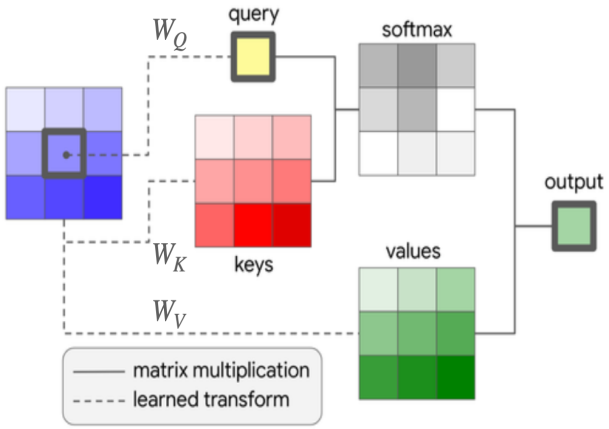

FIGURE 3.3: Representation of the attention mechanism described in (3.3.3). Image taken from [186].

At first sight, the fact that the attention mechanism is a global mechanism unlike that of convolution may not be immediately clear. In fact, even in the description just made, we used a fixed neighborhood of the pixel whose output we want to calculate. If we fix the dimensions of this neighborhood similarly to those of a convolutional kernel, we are replacing the interaction of the pixel of our interest with its neighboring pixels by passing from (3.3.1) to (3.3.4). What makes the attention mechanism global is the opportunity of fixing the neighborhood $\mathcal{N}_d$ of a much larger dimension than would be possible for a convolutional kernel, even up to the spatial dimension of the image/tensor under consideration. In fact, in the case of the attention described in this paragraph, the operations are those of vector-matrix products with the matrices W_Q,W_K,W_V, depending only on the number of input and output channels, while the spatial dependence is due only to the number of times these operations must be repeated for each pixel of the neighborhood. However, this spatial dependence is easily to parallelize as each vector-matrix product does not depend on the others. On the contrary, the convolutional kernel has a complexity closely linked to the size of the neighborhood, which becomes the spatial dimension of the kernel. Hence, the convolution operator requires a tensor-tensor product, that is less parallelizable than attention. In fact, it is rare to find convolutional kernels larger than $7 \times 7 \times C$, while the neighborhoods of the attention mechanisms also have dimensions of 32×32 and beyond.

The attention mechanism described in this section is the simplest one introduced in NLP works. Recent developments led to generalizing the concept

of assigning a weight to a particular pixel based on information deriving from the entire context. For a complete overview of all the attention mechanisms developed in computer vision, we refer the reader to [187]. Anticipating what we will describe in Section 3.4.3, in our work we developed a concept of attention, capable of weighting the pixels during the reconstruction phase of the binary mask, using both spatial and temporal information.

3.4 Proposed model

In this section, we describe and motivate the structure of our model. We use a model resembling a Siamese U-Net consisting of 4 main components:

- Siamese encoders constituted by a pre-trained backbone (see Section 3.4.2).
- Mix and Attention Mask Block (MAMB) and bottleneck mixing block to compose backbone results (see Section 3.4.3).
- Up-sample decoder to refine low resolution results incorporating higher resolution data from the skip connections (see Section 3.4.4).
- Pixel level classifier (see Section 3.4.5).

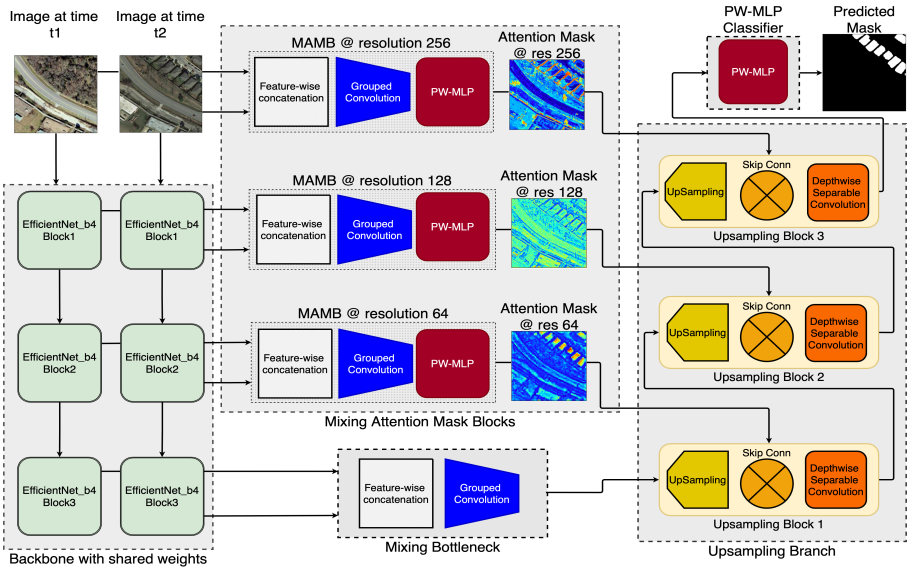

FIGURE 3.4: Siamese U-Net architecture including MAMB.

In what follows, we denote with $X \in \mathbb{R}^{(C \times H \times W)}$ the *reference* tensor (image at time t_1) and with $Y \in \mathbb{R}^{(C \times H \times W)}$ the *comparison* tensor (image at time t_2). C is the number of channels, H is the height, and W the width of the tensors. We omit the batch dimension for the ease of notation. We denote with Conv the convolution operator, with PReLU the Parametric Rectified Linear Unit [188], with IN2d the Instance Normalization [189], and with Sigmoid the sigmoidal activation function.

3.4.1 Model overview

Indicating with f_k the composition of the backbone blocks up to the k^{th} one, the high-level features $X_k = f_k(X)$ and $Y_k = f_k(Y)$ are extracted from each level k of the backbone. These features are used both to compute the resulting output at each level of the U-Net encoder, and to estimate the attention masks. The last backbone block produces the embeddings X_e and Y_e representing the bottleneck inputs.

Every backbone intermediate output pair (X_k, Y_k) is processed by means of the MAMB, producing spatial attention masks M_k. These masks are used as skip-connections and composed in the decoder. The last mixed tensor is obtained by composing (X_e, Y_e).

The decoder consists of a series of up-layers, one for each block of the backbone. Each up-layer increases the spatial dimensions of the tensor received from the previous layer to reach the same resolution of the corresponding skip-connection. Furthermore, the up sampled tensors and the skip-connections are composed to generate the next layer inputs. This composition is the attention mask application to the features obtained from the previous layer.

Finally, the last block of our model classifies each pixel of the obtained tensor through a Pixel-Wise Multi-Layer Perceptron (PW-MLP). The PW-MLP associates to each pixel the probability that it belongs to the anomaly class. Applying a threshold to this tensor we obtain the binary mask of changes. Figure 3.4 depicts the whole architecture of the proposed model.

In the following sections, we describe each component separately.

3.4.2 Siamese encoders with pre-trained backbone

The purpose of the Siamese encoder is to extract features simultaneously from both images in a semantic coherent way. In deep neural networks, training the first layers of the model is sometimes difficult due to the well-known phenomenon of vanishing gradients [131, 190]. To overcome this problem, several tricks have been introduced such as the residual connections of ResNet [45], or the skip connections of the U-Net [11]. However, training deep backbones remains a difficult, time-consuming, or even impossible task to accomplish if the dataset is too small.

For these reasons, pre-trained backbones are often preferred, even in Change Detection problems [137, 141, 144, 147, 151]. The disadvantage of this approach is that the backbones are not always trained on images that are similar to the ones we are dealing with. However, Convolutional Neural Network backbones work by layering information. Low-level features, such as lines, black/white spots, points, edges, can be considered general-purpose being common to all images.

In our intuition, the faced task, that is the comparison between two images I_1 and I_2, can be accomplished by using just the low-level features extracted from the first few layers of a pre-trained backbone. We therefore experimented our architecture with different backbones sliced at different levels, ending up with the EfficientNet backbone [191] pre-trained on the ImageNet dataset [192]. We allowed the training phase to tune also the totality of the backbone parameters. Guided by experiments on our industrial dataset, the EfficientNet backbone

family have been selected due to both its efficacy and its efficiency. Moreover, the resolution reduction in the first EfficientNet layers is sufficiently slow in order to create skip connections of different spatial dimensions.

For completeness, in section 3.5.6 we compare the performances of other backbones.

3.4.3 Mix and Attention Mask Block (MAMB) and bottleneck mixing block

The purpose of this block is to merge the features (X_k, Y_k) extracted from one of the blocks of the Siamese encoder. It creates a mask M_k that is then used as skip connection to refine the information obtained during the up-sampling phase.

The mask we create can also be understood as a pixel-level attention mechanism. The idea of pixel-wise attention has been already studied in [193]. Here we specifically designed a pixel-wise attention mechanism exploiting both spatial and temporal information.

The MAMB can be divided into two sub-blocks: the Mixing block (see Section 3.4.3), and the Pixel level mask generator (see Section 3.4.3).

Mixing block

As the name suggests, in this sub-block we compose the features generated by the k^{th} backbone blocks (X_k, Y_k). To this aim, we observe that the features X_k and Y_k, share both the same shape $C_k,\ H_k,\ W_k$, and the same arrangement in terms of features. This means that the features in channel c of X_k have the same semantic meaning with respect to the corresponding features in channel c of Y_k, being the Siamese encoder weights shared. In view of this observation, we decided to concatenate the tensors X_k and Y_k in the tensor $Z_k \in \mathbb{R}^{2C_k \times H_k \times W_k}$ using the following rule:

$$Z_k^c := \begin{cases} X_k^{c/2} & c \text{ even} \\ Y_k^{(c-1)/2} & c \text{ odd} \end{cases} \quad \forall c \in \{0, ..., 2C_k - 1\}. \tag{3.4.5}$$

To mix the features coming from X_k and Y_k both spatially and temporally, we used a group convolution. By choosing the number of groups equal to C_k we obtain C_k kernels of depth 2 which process the tensor Z_k in pairs of channels. These kernels perform at the same time both spatial and temporal convolution using the cross-correlation between semantically similar features.

The new tensor $Z'_k \in \mathbb{R}^{C_k \times H_k \times W_k}$ is defined as:

$$\begin{aligned} Z'_k = \text{Mix}(X_k, Y_k) := \text{PReLU}[\,\text{IN2d}[\text{Conv}(Z_k,\ ch_{in} = 2C_k, \\ ch_{out} = C_k,\ groups = C_k)]]. \end{aligned} \tag{3.4.6}$$

An illustration of our concatenation strategy, and the following grouped convolution, is reported in Figure 3.5.

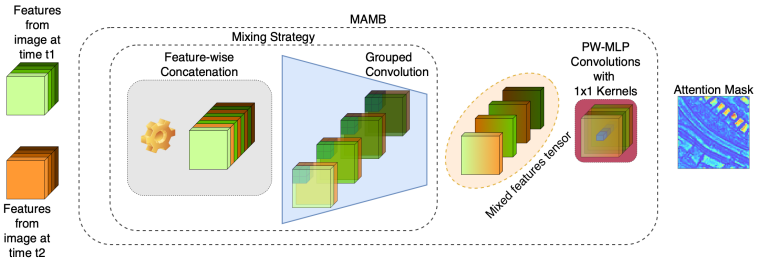

FIGURE 3.5: Visual representation of our mixing strategy and the full MAMB block. In the inner dashed block we highlight the concatenation strategy (3.4.5) and the grouped convolution (3.4.6). These two blocks, when coupled with the PW-MLP, form the MAMB block.

Pixel-level mask generator

By considering fixed the spatial coordinates of a single pixel, it can be seen that the C_k values in the tensor Z'_k contain spatial information related to both times t_1 and t_2. Our idea is to use the PW-MLP in order to process these informations and generate a score that acts as a spatio-temporal attention. To this aim, the PW-MLP is designed to produce a mask tensor $M_k \in \mathbb{R}^{H \times W}$.

PW-MLP

To implement a Pixel-Wise Multi-Layer Perceptron, that is an MLP working on all the channels of one single pixel at a time, we use convolutions with 1×1 kernels. The MLP is composed by N blocks each containing one 1×1 convolution and one activation function. As activation, we used the PReLU, being this able to propagate gradients also on the negative side of the real axis. The last convolution contains just one filter, thus producing a tensor M_k with dimensions 1, H_k, W_k.

The use of 1×1 convolutions to implement an MLP has been already investigated. In [194] this strategy has been used to substitute layers such as convolutions with small, trainable, networks. As pointed out in [194], we have very poor prior information on the latent concepts in pixel vectors. Hence, we have decided to use this universal function approximator to separate different semantic concepts.

The bottleneck mixing block

We applied the tensor mixing strategy reported in Section 3.4.3 to compute the bottleneck of the U-Net like network. More precisely, we compute: $U_e = \text{Mix}(X_e, Y_e)$.

U_e represents the output of the encoder and the input to be processed by the decoder. U_e contains the spatially and temporarily correlated higher level features computed by the backbone. Given that, our intuition is that U_e contains enough information in order to classify each pixel at the bottleneck resolution.

3.4.4 Up-sampling decoder with skip connections

The general $k-$th decoder block takes as input the tensor U_{k+1} of shape $C_{k+1}, H_{k+1}, W_{k+1}$ and a mask M_k of shape $1, H_k, W_k$. Firstly, an up-sampling operation is performed in order to transform U_{k+1} so that its shape matches the one of M_k. We call the up-sampled tensor U'_k. Then, we define U_k with:

$$U_k := \text{PReLU}\left[\text{IN2d}\left[\text{Conv}(U'_k \odot M_k)\right]\right],$$

where we have denoted with the symbol $\odot$ the Hadamard product. This represents the skip connection attention mechanism at the pixel level.

As we already mentioned in Section 3.4.3, U_e contains enough information to classify each pixel at its spatial resolution. By multiplying the mask M_k, we are re-weighting each pixel in order to alleviate the misleading information generated by up sampling.

Notice that, in this Up block we employ the depth-wise separable convolution [195, 196].

3.4.5 Pixel-level classifier

Finally, since the change detection problem is a binary classification problem, we decided to use as last layer a PW-MLP with output classes $\{0, 1\}$ representing respectively normal and changed pixels. With respect to what reported in Section 3.4.3, in this case we used as the last activation layer a Sigmoid function instead of the PReLU, thus enforcing the result of the network to contain values in $[0, 1]$. In this case, the PW-MLP is used as a non-linear classifier which separates pixels in normal and changed classes respectively.

3.5 Experiment Settings and Results

In this section, we present the settings used in our experiments, the achieved results, and the performed ablation study.

3.5.1 Datasets

In order to fairly evaluate our model, and to compare it with other works in the Change Detection field, we used the public aerial building images datasets: LEVIR-CD [137] and WHU-CD [142]. Notice that the task defined by these datasets is particularly close to the task faced by our industrial research. More precisely, in these two datasets the model has to track some specific patters, those corresponding to buildings, and carefully segments the eventually occurred changes.

LEVIR-CD contains 637 pairs of high resolution aerial images. Starting from these images, patch pairs of size 256×256 each have been extracted. After that, the pair instances have been partitioned accordingly to the authors' original indications. This step produced 7120, 1024, and 2048 pair instances for the train, validation, and test dataset, respectively.

WHU-CD contains just one pair of images having resolution 32507×15354 as a crop of a wider geographic area. Following [197], the images have been split into non overlapping patches with resolution 256×256. After that, a randomly partitioning of the dataset have been performed obtaining 5947, 743, and 744 pairs for train, validation, and test respectively.

3.5.2 Loss function and evaluation metrics

As stated in Section 3.4.5, we cast the Change Detection problem in a pixel-wise binary classification setting. In fact, the role of the final PW-MLP block is to output the per-pixel change probability.

Since the reference mask is a binary mask (0 for unchanged pixels, 1 for changed pixels), and since we are comparing probabilities, one loss function that can be used is the Binary Cross Entropy (BCE). It is defined as:

$$\mathcal{L}(G, P) := -\frac{1}{|H| \cdot |W|} \sum_{h \in H, w \in W} g_{h,w} \log(p_{h,w}) + (1 - g_{h,w}) \log(1 - p_{h,w}),$$

where we denoted with G the ground truth mask, with P the model prediction, and with H and W the set of indices relative to height and width.

Notice that the BCE loss function is widely used in other state-of-the-art models such as [144, 152]. In contrast, other researchers implemented more sophisticated loss functions like the one presented in [137]. We decided to use the simpler BCE in order to attribute the improvement in performances to the model and not to an ad hoc built-in loss function. For completeness, in Section 3.5.7 we report additional experiments with other loss functions.

To evaluate the performances achieved by our model, we calculated the *Precision (PR)*, *Recall (RC)*, *F1 score (F1)*, *Intersection over Union (IoU)* and *Overall Accuracy (OA)* with respect to the change class, as defined below:

$$
\begin{aligned}
Pr &:= \frac{TP}{TP + FP}, \\
Rc &:= \frac{TP}{TP + FN}, \\
F1 &:= \frac{1}{Pr^{-1} + Rc^{-1}}, \\
IoU &:= \frac{TP}{FN + FP + TP}, \\
OA &:= \frac{TP + TN}{FN + FP + TP + TN},
\end{aligned}
$$

where TP, TN, FP, FN are computed on the change class, and represent the true positives, true negatives, false positives, and false negatives respectively. To retrieve the change mask we applied a 0.5 threshold to the output mask.

3.5.3 Implementation details

We implemented our model using PyTorch [198], and we trained it on an NVIDIA GeForce RTX 2060 6 GB GPU. As described in Section 3.4.2, we selected the first four blocks of the EfficientNet version $b4$ backbone pretrained on the ImageNet dataset. All other weights of the model have been initialized randomly.

As optimization algorithm, we adopted AdamW [199]. To optimize its hyperparameters, i.e. learning rate, weight decay, and `amsgrad` variant, and also to verify the robustness of our model with respect to the choice of these parameters, we firstly run a Hyper-Parameters-Optimization task for each dataset using the package Neural Network Intelligence (NNI) [200]. After this, we fixed the learning rate to $3 \cdot 10^{-3}$, and the weight decay to $9 \cdot 10^{-3}$, for the LEVIR-CD dataset. Moreover, we fixed the learning rate to $2 \cdot 10^{-3}$, and the weight decay to $8 \cdot 10^{-3}$, for the WHU-CD dataset. For both datasets, `amsgrad` have been set to `False`. An example of the HPO procedure is reported in section 3.5.7. Due to computational resource limitations, no other hyperparameters have been tuned. We left as future work the exploration of network architecture search techniques (NAS) applied to Change Detection tasks.

To dynamically adjust the learning rate during the training, we adopted the cosine annealing strategy as described in [201], but avoiding the warm restart.

Since aerial images are spatially registered, we applied the geometric data augmentation operators simultaneously to the reference/comparison images and their associated ground-truth mask. Also, non-geometric augmentations have been applied independently on the reference and the comparison images.

The applied geometric augmentations are Random Flip on both X and Y axes, and Random Rotation with free degree. Moreover, the applied non-geometric augmentations are Gaussian Blur and Random Brightness/Contrast change. To achieve all the adopted augmentations, we used the Albumentations library [202].

Finally, due to the limited GPU memory capacity and computational power, we fixed the batch size to 8, and trained for just 100 epochs.

3.5.4 Comparison with state-of-the-art models

To demonstrate the effectiveness of our approach, we compared our results with those reported in [144, 152]. As baseline, we used the three models presented in [146]. Moreover, to compare our model with other works adopting both spatial and channel attention mechanisms, we dealt with [137, 141, 148, 168]. Finally, given the success achieved by the Transformers applied to the computer vision field, we also compared our results with those obtained in [144, 152].

The results reported in Table 3.1 and Table 3.2 show the superior performance of our model on the LEVIR-CD and WHU-CD building change detection datasets.

TABLE 3.1: Performance metrics on the LEVIR-CD dataset. To improve results readability, we adopted a color ranking convention to represent the First, Second, and Third results respectively. The metrics are reported in percentage.

LEVIR-CD					
Model	Pr	Rc	F1	IoU	OA
FC-EF [146]	86.91	80.17	83.40	71.53	98.39
FC-Siam-diff [146]	89.53	83.31	86.31	75.92	98.67
FC-Siam-conc [146]	91.99	76.77	83.69	71.96	98.49
DTCDSCN [148]	88.53	86.83	87.67	78.05	98.77
STANet [137]	83.81	91.00	87.26	77.40	98.66
IFNet [141]	94.02	82.93	88.13	78.77	98.87
SNUNet [168]	89.18	87.17	88.16	78.83	98.82
BIT [144]	89.24	89.37	89.31	80.68	98.92
Changeformer [152]	92.05	88.80	90.40	82.48	99.04
TinyCD	92.68	89.47	91.05	83.57	99.10

The baseline models FC-Siam-diff and FC-Siam-conc [146] are the architectures most similar to ours. With respect to these two baseline models, we increased the F1 score by 4.73 points on LEVIR-CD, and by more than 20 points on the WHU-CD. With respect to the best model we found in the literature [152], our performance increment on the LEVIR-CD dataset is more limited. However, as we can see from Table 3.3, our model is 146 times smaller.

In view of these results, we can conclude that our model, despite the lower complexity and the lower number of employed parameters, is very effective on the buildings Change Detection task. Moreover, having not used any global attention

TABLE 3.2: Performance metrics on the WHU-CD dataset. To improve results readability, we adopted a color ranking convention to represent the First, Second, and Third results respectively. The metrics are reported in percentage.

	WHU-CD				
Model	Pr	Rc	F1	IoU	OA
FC-EF [146]	71.63	67.25	69.37	53.11	97.61
FC-Siam-diff [146]	47.33	77.66	58.81	41.66	95.63
FC-Siam-conc [146]	60.88	73.58	66.63	49.95	97.04
DTCDSCN [148]	63.92	82.30	71.95	56.19	97.42
STANet [137]	79.37	85.50	82.32	69.95	98.52
IFNet [141]	96.91	73.19	83.40	71.52	98.83
SNUNet [168]	85.60	81.49	83.50	71.67	98.71
BIT [144]	86.64	81.48	83.98	72.39	98.75
TinyCD	91.72	91.76	91.74	84.74	99.34

TABLE 3.3: Parameters, complexity, and performance comparison. The metrics are reported in percentage, parameters in Millions (M), and complexity in GFLOPs (G).

Model	Param (M)	Param ratio	FLOPs (G)	LEVIR-CD F1	WHU-CD F1
DTCDSCN [148]	41.07	146.67	7.21	87.67	71.95
STANet [137]	16.93	60.46	6.58	87.26	82.32
IFNet [141]	50.71	181.10	41.18	88.13	83.40
SNUNet [168]	12.03	42.96	27.44	88.16	83.50
BIT [144]	3.55	12.67	4.35	89.31	83.98
Changeformer [152]	41.02	146.50	N.D.	90.40	N.D.
TinyCD	0.28	1	1.45	91.05	91.74

mechanism, we have a confirmation of our intuitions: in the faced Change Detection task, low level information is sufficient to reach high-quality results. Also, the information contained in each single pixel at different resolutions, is very rich and can be exploited to effectively classify changes.

In Figure 3.6, a visual/qualitative comparison between the masks created by our model, and those created by BIT [144] on the LEVIR-CD test dataset, is reported. Both models perform well, and we end up our analysis by conjecturing that the performance difference reported in Table 3.1 and Table 3.2

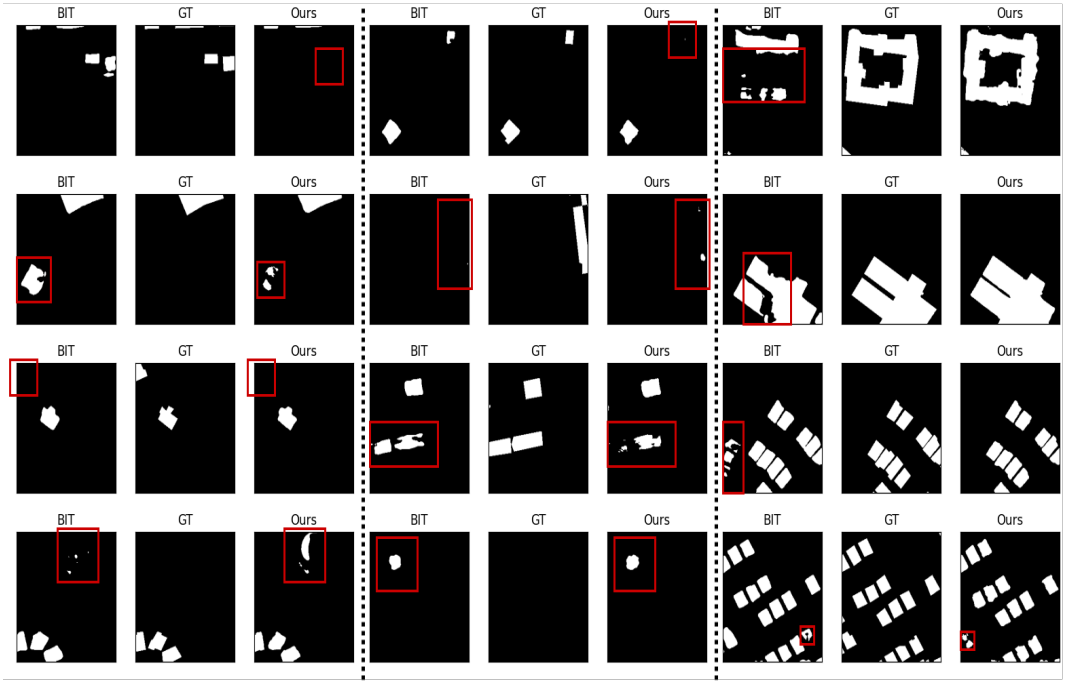

FIGURE 3.6: Visual comparison between outputs obtained by our model and BIT. We highlighted with red bounding boxes those regions containing significant differences between the ground-truth and the generated masks.

are more related to missing or hallucinated objects, than region quality issues. Nevertheless, we can find some examples were there are significant differences between the ground truth masks (GT) and those created by the two models. In Figure 3.6, it is interesting to note that there are examples where both models fail similarly in the same regions, despite the two models being based on very different approaches (local versus global).

3.5.5 Ablation study

In this section we describe the adopted ablation study steps and the achieved results.

Backbone dimension and final PW-MLP

The first ablation study we conducted concerns the size of the backbone and the use of the final MLP. Regarding the backbone size, we considered both the whole EfficientNet-b4 except the final classifier, and a sliced version of the EfficientNet-b4 network including just the first 3 blocks. Moreover, to assess the effectiveness of the final classification PW-MLP block, we considered both the architecture including it, and the one that produces its output directly from the last up-sampling block by forcing it to output just one channel. The results shown in Table 3.4 confirm our intuition on low-level features. In fact, our solution with the sliced backbone and final PW-MLP, turns out to be the one with the best performances on both datasets. Furthermore, we note that, to get the best performances, the backbone slicing and PW-MLP classifier must

be coupled. In fact, on LEVIR-CD the model containing just the backbone slicing shows poor performances, while the use of the PW-MLP classifier helps the full backbone architecture to improve the quality of the segmentations. In contrast, on the WHU-CD the architecture with sliced backbone and the PW-MLP classifier obtains better scores than the one with full backbone but without PW-MLP, remaining the performances of the latter still unsatisfactory and far from those obtained by our model.

TABLE 3.4: Performance comparison between versions of our model including and excluding the backbone slicing and the PW-MLP classifier.

LEVIR-CD						
Model	Precision	Recall	F1 score	IoU	Accuracy	Params
Full w/o MLP	83.05	94.00	88.19	78.88	98.71	17740598
Full w MLP	92.65	89.26	90.92	83.36	99.09	17743288
Sliced w/o MLP	46.15	94.52	62.02	44.95	94.10	282438
Sliced w MLP	92.68	89.47	91.05	83.57	99.10	285128
WHU-CD						
Model	Precision	Recall	F1 score	IoU	Accuracy	Params
Full w/o MLP	43.08	88.12	57.87	40.72	94.91	17740598
Full w MLP	91.00	92.14	91.57	84.45	99.32	17743288
Sliced w/o MLP	76.16	89.05	82.10	69.64	98.84	282438
Sliced w MLP	91.72	91.76	91.74	84.74	99.34	285128

Comparison with other simple mixing strategy

In Table 3.5, we compare our mixing strategy, described in Section 3.4.3, with other widely used feature fusion blocks. We tested the following alternatives:

- subtraction, both in the bottleneck and in skip connections;
- concatenation + convolution, both in the bottleneck and in skip connections.

We selected these two alternatives since our mixing strategy can be seen as a generalization of the pixel-wise subtraction. In fact, if we initialize all of our 2-depth kernels with the "central" weights to 1 and -1, and all the rest to 0, we have the standard subtraction. However, our mixing block Section 3.4.3 is fully trainable with the spirit of feature re-use [203]. Moreover, concatenation + convolution can be seen as generalization of our mixing block. However, the number of trainable parameters to be tuned for this mixing block is much bigger

than ours. More precisely, the number of parameters in our mixing block is $c(2 \cdot k_h \cdot k_w)$, where c is the number of channels, k_h, k_w are the convolutional kernel sizes. By comparison, a convolution working on the concatenated feature tensors contains $c(2c \cdot k_h \cdot k_w)$ parameters. The parentheses are highlighting the size of each kernel and the number of kernels.

TABLE 3.5: Performance comparison between the model with our mixing strategy, subtraction, and concatenation + convolution (C+C) respectively.

LEVIR-CD							
Model type	Pr	Rc	F1	IoU	OA	Param. tot.	GFLOPs ±
Subtraction	92.13	89.41	90.75	83.07	99.07	282939	1.43 (−1.4%)
C+C	92.55	89.61	91.06	83.59	99.10	368468	1.75 (+20.7%)
TinyCD	92.68	89.47	91.05	83.57	99.10	285128	1.45
WHU-CD							
Model type	Pr	Rc	F1	IoU	OA	Param. tot.	GFLOPs ±
Subtraction	90.10	91.55	90.82	83.19	99.26	282939	1.43 (−1.4%)
C+C	92.19	91.25	91.72	84.71	99.34	368468	1.75 (+20.7%)
TinyCD	91.72	91.76	91.74	84.74	99.34	285128	1.45

In Table 3.6 the results of a more detailed study on mixing strategies are reported. We alternated the use of subtraction/concatenation + convolution with our respective proposal to mix the features in the bottleneck/skip connections.

TABLE 3.6: Evaluation of subtraction and concatenation + convolution mixing strategies. We reported F1 score for the two datasets LEVIR-CD (F1-L) and WHU-CD (F1-W). We used ✗to indicate where we changed our proposed option with subtraction or concatenation + convolution. In contrast, ✓represents our bottleneck mixing block or MAMB.

(A) Subtraction

Mix	Skip	F1-L	F1-W	Param
✗	✗	90.75	90.82	282939
✓	✗	90.75	91.51	284004
✗	✓	90.71	89.58	284063
✓	✓	91.05	91.74	285128

(B) Concatenation+Convolution

Mix	Skip	F1-L	F1-W	Param
✗	✗	91.06	91.72	368468
✓	✗	91.06	91.08	313028
✗	✓	90.90	91.71	340568
✓	✓	91.05	91.74	285128

The obtained results confirm that our proposal can be considered an effective generalization of the subtraction, with little impact on the size and complexity

of the model. On the other hand, the overhead introduced by the concatenation + convolution mixing strategy, seems to produce little differences in terms of performance.

Impact of skip connection with MAMB

To quantitatively confirm the usefulness of the skip connections, we trained a model without them and compared the achieved results in Table 3.7.

TABLE 3.7: Performance comparison between the model with/without skip connections on both datasets LEVIR-CD and WHU-CD.

LEVIR-CD					
Model type	Pr	Rc	F1	IoU	OA
No Skip	92.35	88.50	90.38	82.45	99.04
Skip	92.68	89.47	91.05	83.57	99.10
WHU-CD					
Model type	Pr	Rc	F1	IoU	OA
No Skip	90.56	89.77	90.16	82.09	99.22
Skip	91.72	91.76	91.74	84.74	99.34

All the metrics confirm the beneficial effects of skip connections in the model. Figure 3.7 we reports an example of the intermediate masks that our model creates in the skip-connections.

The mask created with the MAMB block at resolution 64 highlights the objects that must be tracked (red pixels). The intermediate mask at resolution 128 acts more like an edge detector. Finally, the mask at resolution 256, obtained applying the MAMB block directly to the original images I_1 and I_2, distinguishes between object classes like buildings and street (dark blue), vegetation (light green), and shadows (red). The ability to highlight shadows is very effective since it helps the model to detect objects and to refine their edges.

Channel-wise MLP vs CycleMLP

As reported in Section 3.2.4, several MLP blocks have recently been studied with the intent of incorporating both spatial and channel-specific information. As previously described, we used the MLPs only along the channels in the final classifier, and coupled to our mixing strategy in the MAMB blocks to obtain space-time correlation. We then decided to deal with the CycleMLP block proposed in [177]. The results reported in Table 3.8 suggest the superiority of our proposed use of MLPs compared to that proposed in [177]. A heuristic explanation for these results can be the following: the MLP blocks proposed in [177] have shown to obtain excellent performances when they are used to

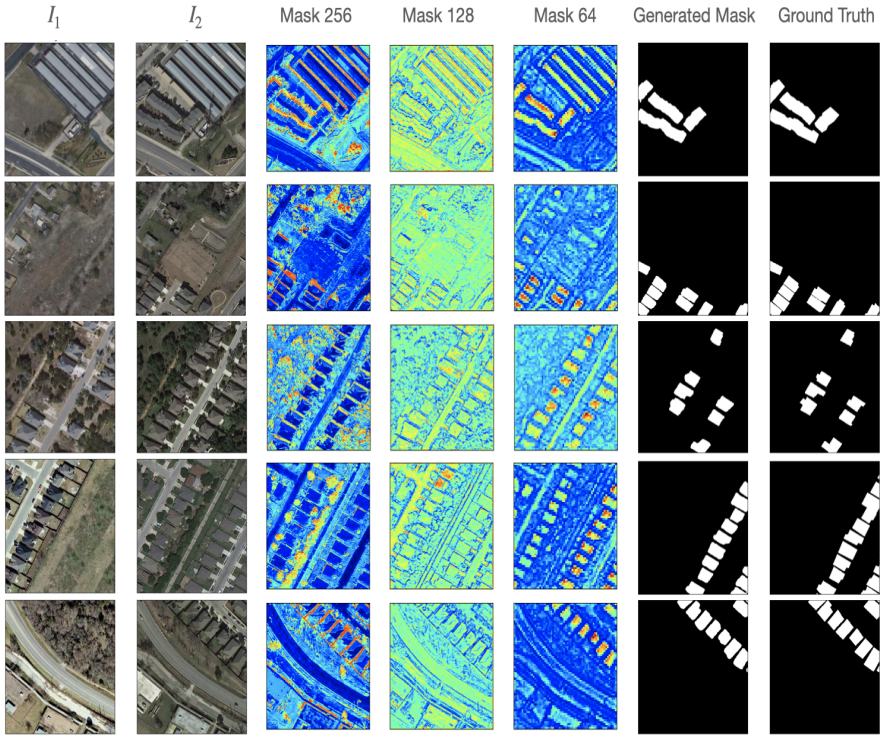

FIGURE 3.7: Visualization of the intermediate masks at different resolutions and the final binary mask for one example image pair.

construct a hierarchical architecture to generate pyramidal features. This makes us think that the advantage of CycleMLPs may be more significant when the features are more refined than the low-level features we use.

3.5.6 Backbones comparison

We report the results obtained by varying the backbone adopted in the model. In each backbone we decided to select all the initial blocks up to the first having spatial resolution 32x32. Due to the different compositions of the considered networks, the final size of the model changes in the range starting from a minimum of 32 thousands parameters up to 1.3 millions.

Table 3.9 shows that the results obtained are stable from the performances point of view. The backbones of the EfficientNet family appear to be, in accordance with the experiments on our proprietary dataset, those that achieve the best performances. However, the other backbone types also produce comparable results making our approach:

- robust with respect to the backbone used;
- flexible with respect to the required size and computational complexity.

In this comparison we have not considered Transformer-type backbones such as [204, 205]. The reason for this choice lies in the fact that the philosophy of the Transformers is a global philosophy, as opposed to the blocks we propose which are instead local. An integration of these two philosophies will be the subject of future works.

TABLE 3.8: Performance comparison between MLP and CycleMLP [177] on LEVIR-CD and WHU-CD. We used ✗to indicate experiments where we changed our proposed block with a CycleMLP one, while ✓represents our proposed architecture.

LEVIR-CD							
Skip	Class.	Pr	Rc	F1	IoU	OA	Param. tot.
✗	✓	92.47	88.48	90.43	82.53	99.04	309300
✗	✗	92.45	88.49	90.42	82.52	99.04	314542
✓	✗	92.58	88.96	90.73	83.04	99.07	290370
✓	✓	92.68	89.47	91.05	83.57	99.10	285128
WHU-CD							
Skip	Class.	Pr	Rc	F1	IoU	OA	Param. tot.
✗	✓	89.76	89.06	89.41	80.85	99.16	309300
✗	✗	92.25	90.51	91.37	84.12	99.32	314542
✓	✗	90.20	85.84	87.96	78.52	99.06	290370
✓	✓	91.72	91.76	91.74	84.74	99.34	285128

TABLE 3.9: Comparison of different backbones on LEVIR-CD dataset

LEVIR-CD						
Backbone	Precision	Recall	F1 score	IoU	Accuracy	Params
mobilenetv2	90.95	86.43	88.63	79.59	98.87	38798
mobilenetv3large	90.56	85.98	88.21	78.91	98.82	32886
resnet18	92.15	87.43	89.72	81.37	98.98	707894
efficientnetb0	92.18	87.96	90.02	81.85	99.00	79480
efficientnetb1	92.17	88.92	90.51	82.67	99.05	122092
efficientnetb2	92.13	89.26	90.68	82.94	99.06	148040
efficientnetb3	92.40	89.54	90.95	83.40	99.09	178716
efficientnetb4	92.68	89.47	91.05	83.57	99.10	285128
mnasnet13	91.95	88.17	90.02	81.86	99.00	97262
densenet121	92.13	87.97	90.00	81.83	99.00	1364790

3.5.7 Hyperparameters' tuning

One of the advantages of using limited computational complexity models, is being able to fine-tune hyperparameters using relatively few computational resources, and in a reasonable time, from an industrial point of view. In our experiments we tune the learning rate, the weight decay, and the usage of the *amsgrad* strategy. The framework used to run the experiments and optimize the hyperparameters is NNI [200].

Since we execute only 100 epochs per run, we chose a higher learning rate range $(10^{-3}, 4 \cdot 10^{-3})$, in order to explore whether a higher than standard learning rate leads to faster model convergence. As for the weight decay, we follow a conservative choice by setting the range between 10^{-2} and $8 \cdot 10^{-3}$. We also test other simple loss functions for model training such as Mean Square Error (MSE), Intersection over Union (IoU) and a combination of IoU and BCE.

In Figure 3.8 we show the various combinations of hyperparameters explored in a batch of 30 experiments, and the relative performances on the LEVIR-CD validation set. Analyzing the results, we note that BCE and MSE, regardless of the other parameters, obtain superior performance compared to the IoU. In addition, the BCE + IoU combination, although better than IoU, also scores lower than the BCE and MSE. Regarding the other hyperparameters, as can be seen in particular from Figure 3.9, our model obtains robust performances with respect to all the tested combinations. Finally, we note that in the conducted experiments, BCE has lower variance in terms of F1 score with respect to the choices of the other hyperparameters. This represents another motivation for us to chose BCE as loss function.

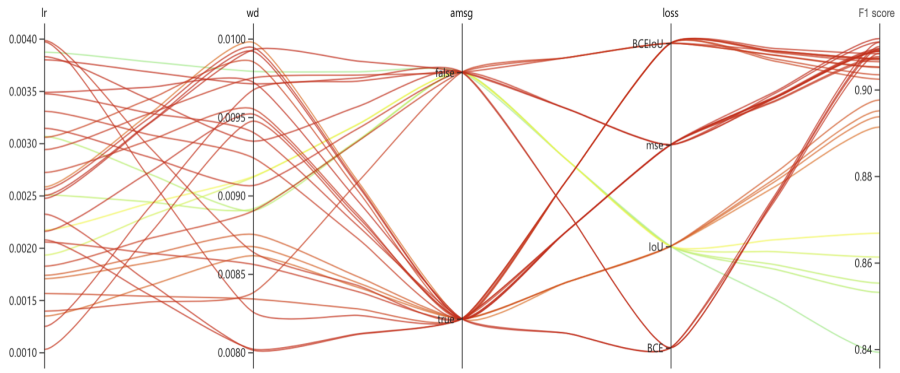

FIGURE 3.8: Different combination of parameters and their impact on the F1 score on the LEVIR-CD dataset.

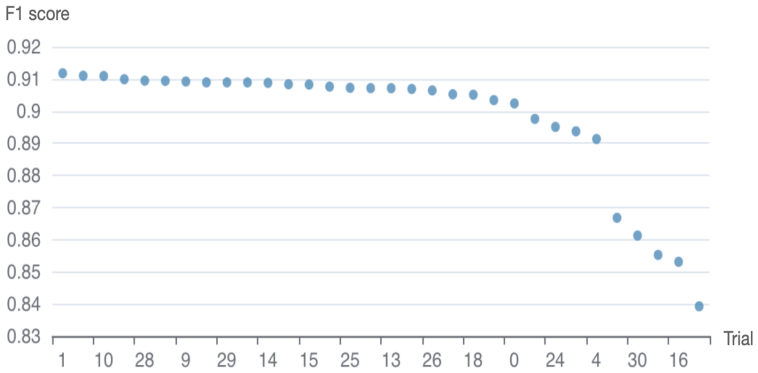

FIGURE 3.9: Behavior of the final F1 score in the different experiments conducted to tune the hyperparameters. The drop in the F1 score is due to the use of IoU as loss function.

3.6 Discussion

In CD problems, the model must be able to compare two input images, highlighting the changes that occurred. The model we presented in this work, TinyCD, exploits low-level features by comparing them and classifying pixels to obtain a binary map of detected changes. Despite the lower computational complexity and the reduced number of parameters, the results achieved place TinyCD among the CD state-of-the-art models.

Among others, the adoption of low-level features also allows to reduce the depth of the included backbone part, thus reducing the overall depth of the model. In this way, the resulting network is wider than deeper, and this allows to reduce the gradient confusion, improving the training efficiency [206]. Moreover, smaller models are easier and faster to train, allowing a more effective and affordable HPO phase. In industrial applications this is a desirable feature, since the model can be retrained according to the necessity of the costumers.

We have shown that an effective way to generate the output mask is to process low-level backbone features with a PW-MLP block, facing the change-detection task as a per-pixel classification problem. Moreover, the results of the ablation study reported in Table 3.4, show that low-level backbone features, and the final PW-MLP classifier, perform best when coupled.

The mixing strategy that we have introduced, and the MAMBs derived from it, leverage the structure of the Siamese networks. Indeed, the features extracted from the two branches of the network share the semantics associated to the single channels/filters. Our mixing strategy takes this as an advantage, efficiently comparing those corresponding features in a spatio-temporal manner. In the case of MAMBs, the low-level features are exploited to form the attention masks. These masks show how the low level features are able, at different levels, to help the model to pay attention to details of different meanings.

3.7 Experimental Results on Industry Machines

In this section, we show the application of TinyCD on industrial machines. Notice that the results presented here must be considered just a proof of concepts regarding the applicability of TinyCD to the original industrial scenario. It is simplistic to reduce the solution to the problem presented in the introduction of Chapter 3 to the application of TinyCD to a specific dataset.

The dataset for this experiment have been collected by placing cameras directly on working machines. Figure 3.10 reports examples of different acquisition points.

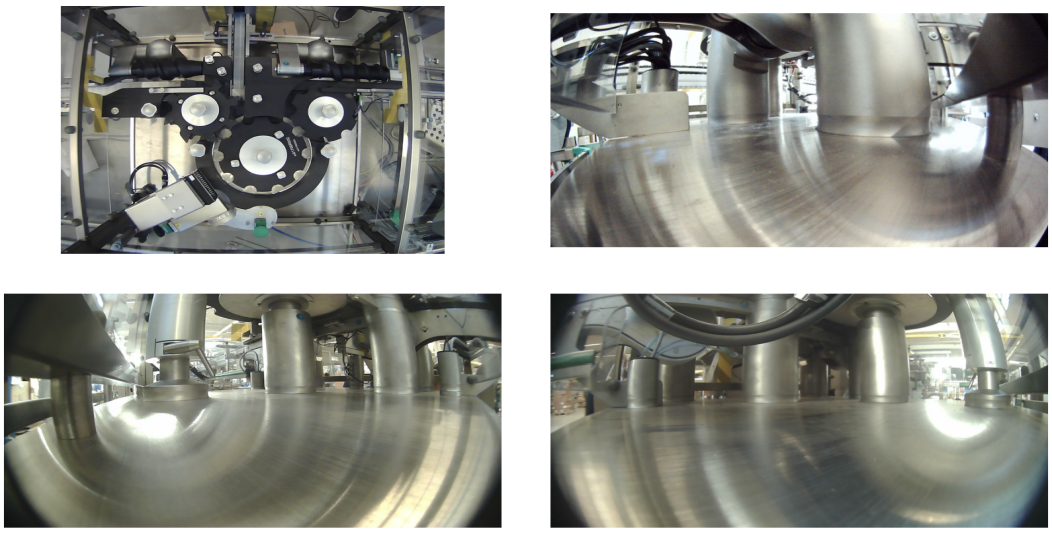

FIGURE 3.10: Examples of different points of view of an industrial machine used in this experiment.

To simulate anomalous and unwanted objects in the scene, objects like pills, pharma packages, pipettes, and mechanical tools, have been collected and then used to create augmented images containing anomalies. To evaluate the ability of the model to track unwanted changes also caused by objects not included in the training set, we used different classes of objects for training and testing. Other augmentations have been adopted to address image variations like light conditions, perspective, rotation, deformations and colour changes; these have been employed to mimic the working conditions, and to improve the generalization ability of the model.

The results reported in Figure 3.11 and Figure 3.12 confirm the effectiveness of TinyCD also in the industrial context. Since the sets of objects used during training and testing are disjointed, we can also affirm that our model has successfully learned to highlight unadmitted changes by comparing the two images without using specific patterns of unwanted objects.

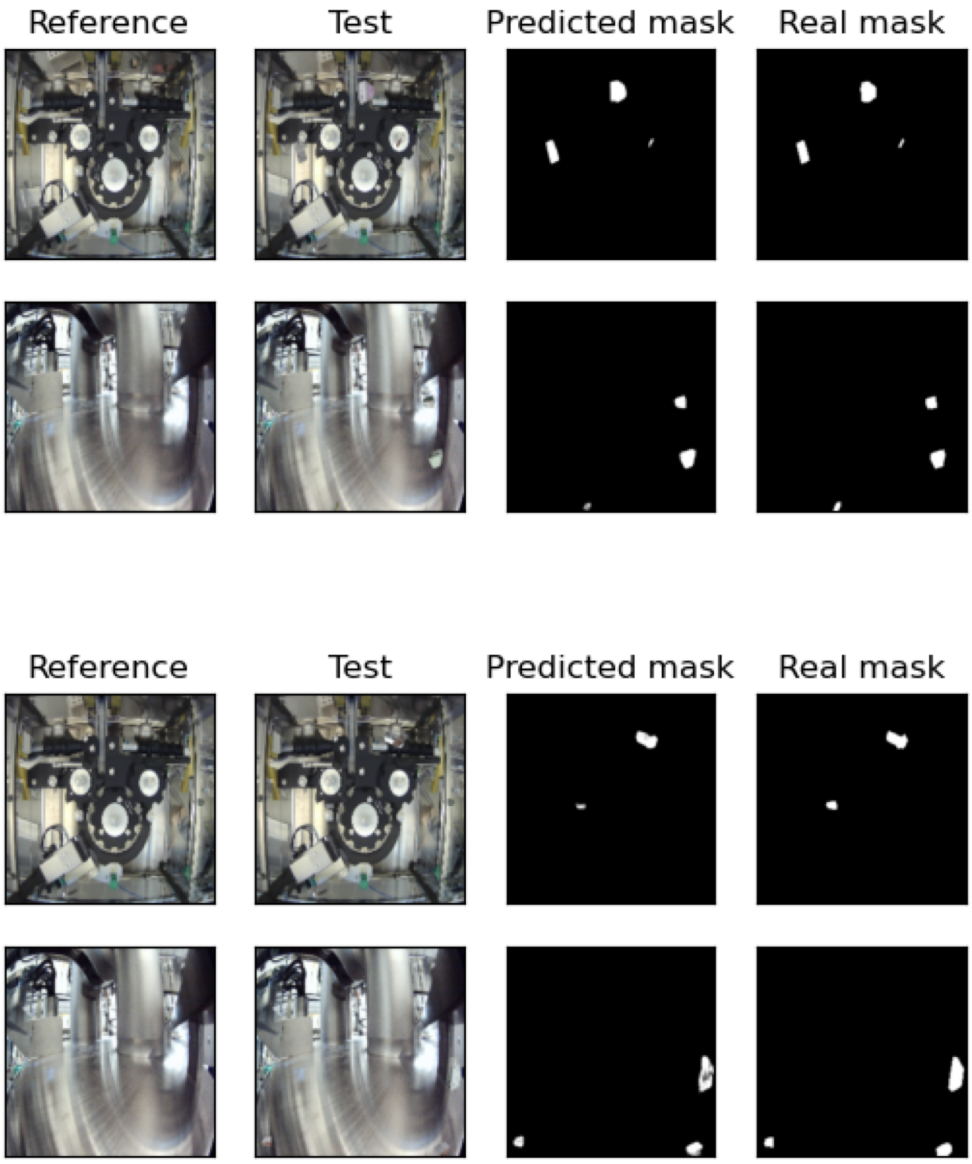

FIGURE 3.11: Qualitative results on industrial machines.

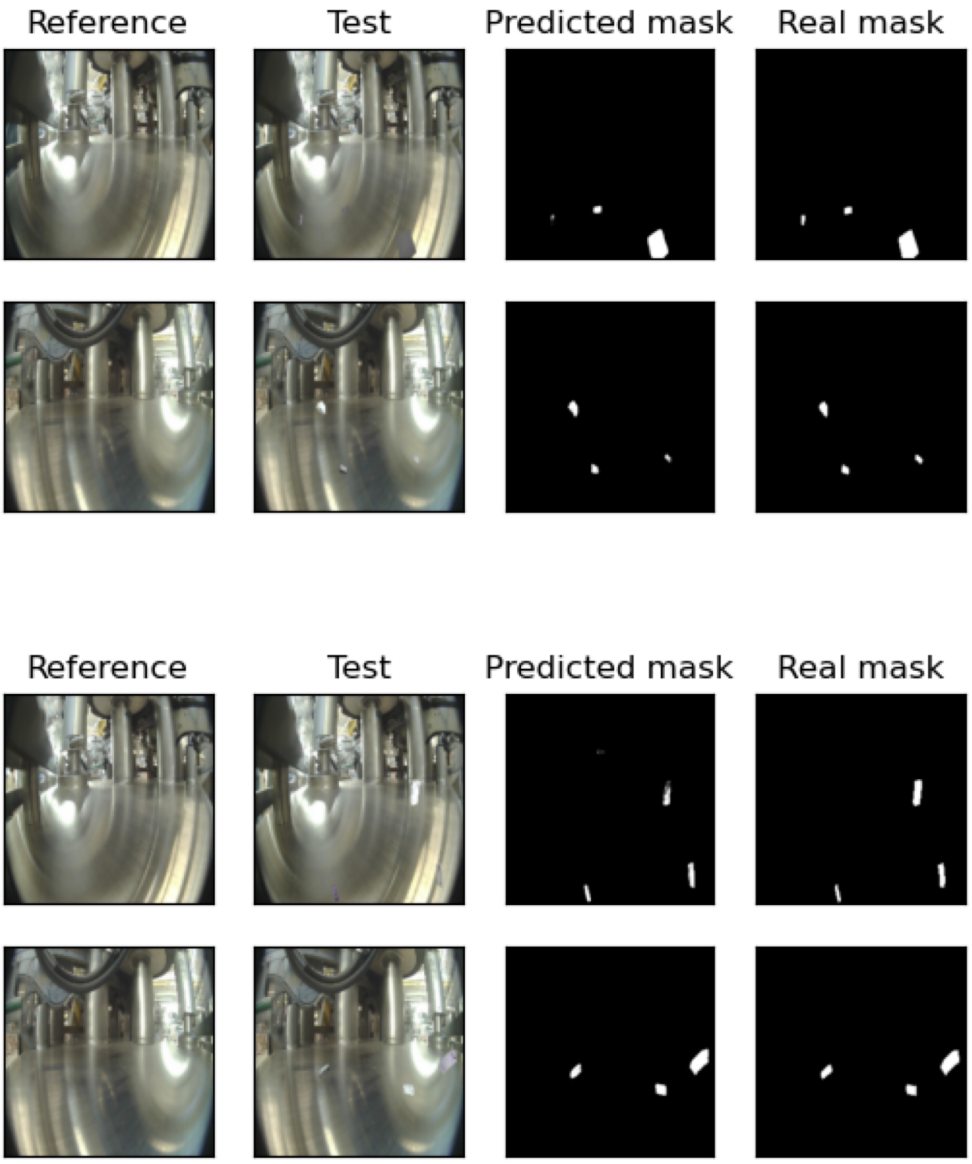

FIGURE 3.12: Qualitative results on industrial machines.

Chapter 4

Conclusions and future works

In Chapter 2 we have introduced and studied the family of Fourier-based metrics in the context of image processing. In this setting, we have extended the definition of this family to deal with measures that do not satisfy the moment equality requirement. Moreover, we have explicitly derived the equivalence constants that relate the Fourier-based metrics and the Wasserstein distance on the discrete setting. We have computed the equivalence constants in the general case without imposing restrictions on the family of measures we are considering, and hence they are influenced by the periodicity of the Fourier transform which determines the dependency from the measure of the support of the probability measures. The first line of research could be a characterization of the family of measures on which Fourier-based metrics and Wasserstein metrics that have support space-free equivalence constants.

The numerical results we have presented show that in applications the relationship between Fourier-based and Wasserstein metrics is stricter than the theoretical one. This means that Fourier-based metrics can be used in a wide range of applications with similar results, especially where fast computations are needed, for example, in Clustering algorithms. Typically, these algorithms require to calculate the reciprocal distances of numerous samples, and each sample could have large spatial dimensions. The statistical study of the results obtained with Wasserstein and Fourier-based metrics, together with a theoretical study of the guarantees that Fourier-based metrics could provide, would have a considerable impact on real applications on large datasets, drastically reducing computational times while maintaining statistically consistent results. Even in the field of Deep Learning, where models are trained to match probability measures, the equivalence between Fourier-based metrics and Wasserstein distances could make interesting contributions. In fact, given the greater ease of use, both computationally and theoretically, Fourier-based metrics can be the right tool to investigate how the weak topology leads Deep Learning models to have better results.

In Chapter 3, guided by our industrial needs, we have proposed TinyCD, a convolutional change-detection Siamese U-Net-like model. TinyCD exploits low-level features by comparing and classifying them to obtain a binary map of detected changes. We investigate the ability of PW-MLP blocks in extracting features and propose a new spatio-temporal features fusion strategy. Combining these two elements, we introduce MAMB, a mixing and attention mask block that we employ to create skip connection masks. The resulting architecture

meets our industrial requirements in terms of computational complexity and deployable on-edge devices.

We tested our model on public change detection datasets containing aerial images acquired at two different times. Furthermore, we compared the achieved results with state-of-the-art models proposed in the change detection literature. Our tests demonstrated that our model performs comparably or better than the current state-of-the-art models, remaining at the same time the smaller and faster one.

Notice that the ideas employed in this work can be also applied to other application domains. For this reason, we will investigate the application of MAMB and PW-MLP blocks to tasks such as anomaly detection, surveillance, and semantic segmentation.

In all the experiments we performed, we observed our model learning some domain-specific patterns. Despite being this an advantage allowing the model to better deal with the faced task, this is also a limitation because it reduces the model's ability to adapt to new scenarios using fine-tuning. Furthermore, in the two datasets taken into consideration, and also in our industrial case study, the images I_1 and I_2 are spatially registered. This allowed the successful usage of low-level features without assessing global feature relationships. In different contexts, where the images undergo large spatial shifts, this local approach can show worse performances with respect to more global approaches like vision transformers [144, 152]. In future works, we plan to investigate solutions to those limitations to extend the applicability of TinyCD. We also left as a future work an extensive study on how our low-level local approach can be beneficial for training and performances in other areas besides that of Change Detection. Moreover, to be able to extend our approach even in those contexts where global features play a fundamental role, we would like to explore multibranch models in which one branch works on local features and one on global features.

List of Figures

List of Tables

Bibliography

[1] R. Gonzalez and R. Woods, *Digital Image Processing*. Prentice-Hall, Inc., 2006.

[2] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks*, vol. 32, pp. 323–332, 2012.

[3] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars," *NIPS 2016 Deep Learning Symposium*, 2016.

[4] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. M. Paixao, F. Mutz *et al.*, "Self-driving cars: A survey," *Expert Systems with Applications*, vol. 165, p. 113816, 2021.

[5] D. Mantegazza, J. Guzzi, L. M. Gambardella, and A. Giusti, "Vision-based control of a quadrotor in user proximity: Mediated vs end-to-end learning approaches," in *International Conference on Robotics and Automation*. IEEE, 2019, pp. 6489–6495.

[6] M. Ferri, D. Mantegazza, E. Cereda, N. Zimmerman, L. M. Gambardella, D. Palossi, J. Guzzi, and A. Giusti, "Training lightweight cnns for human-nanodrone proximity interaction from small datasets using background randomization," *arXiv preprint arXiv:2110.14491*, 2021.

[7] I. Song, H.-J. Kim, and P. B. Jeon, "Deep learning for real-time robust facial expression recognition on a smartphone," in *International Conference on Consumer Electronics*. IEEE, 2014, pp. 564–567.

[8] I. Adjabi, A. Ouahabi, A. Benzaoui, and A. Taleb-Ahmed, "Past, present, and future of face recognition: A review," *Electronics*, vol. 9, no. 8, p. 1188, 2020.

[9] Z. Rezaei, "A review on image-based approaches for breast cancer detection, segmentation, and classification," *Expert Systems with Applications*, vol. 182, p. 115204, 2021.

[10] M. Arif, F. Ajesh, S. Shamsudheen, O. Geman, D. Izdrui, and D. Vicoveanu, "Brain tumor detection and classification by mri using biologically inspired orthogonal wavelet transform and deep learning techniques," *Journal of Healthcare Engineering*, vol. 2022, 2022.

[11] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015, pp. 234–241.

[12] D. Carrera, F. Manganini, G. Boracchi, and E. Lanzarone, "Defect detection in sem images of nanofibrous materials," *Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 551–561, 2016.

[13] T. Czimmermann, G. Ciuti, M. Milazzo, M. Chiurazzi, S. Roccella, C. M. Oddo, and P. Dario, "Visual-based defect detection and classification approaches for industrial applications—a survey," *Sensors*, vol. 20, no. 5, p. 1459, 2020.

[14] A. Kumar, "Computer-vision-based fabric defect detection: A survey," *Transactions on Industrial Electronics*, vol. 55, no. 1, pp. 348–363, 2008.

[15] R. Szeliski, *Computer vision: algorithms and applications*. Springer Nature, 2022.

[16] N. C. Thompson, K. Greenewald, K. Lee, and G. F. Manso, "The computational limits of deep learning," *arXiv preprint arXiv:2007.05558*, 2020.

[17] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1, pp. 89–97, 2004.

[18] D. B. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on pure and applied mathematics*, 1989.

[19] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, "An algorithm for minimizing the mumford-shah functional," in *International Conference on Computer Vision*. IEEE, 2009, pp. 1133–1140.

[20] M. Monteiro, L. Le Folgoc, D. Coelho de Castro, N. Pawlowski, B. Marques, K. Kamnitsas, M. van der Wilk, and B. Glocker, "Stochastic segmentation networks: Modelling spatially correlated aleatoric uncertainty," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 756–12 767, 2020.

[21] J. J. Shen, "A stochastic-variational model for soft mumford-shah segmentation," *International Journal of Biomedical Imaging*, vol. 2006, 2006.

[22] A. Blake, P. Kohli, and C. Rother, *Markov random fields for vision and image processing*. MIT press, 2011.

[23] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *International Conference on Computer Vision*, vol. 1. IEEE, 2001, pp. 105–112.

[24] F. Yi and I. Moon, "Image segmentation: A survey of graph-cut methods," in *International Conference on Systems and Informatics*. IEEE, 2012, pp. 1936–1941.

[25] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.

[26] W. Wang, J. Shen, and L. Shao, "Consistent video saliency using local gradient flow optimization and global refinement," *Transactions on Image Processing*, vol. 24, no. 11, pp. 4185–4196, 2015.

[27] N. Ray and S. T. Acton, "Motion gradient vector flow: An external force for tracking rolling leukocytes with shape and size constrained active contours," *Transactions on Medical Imaging*, vol. 23, no. 12, pp. 1466–1478, 2004.

[28] G. Aubert, P. Kornprobst, and G. Aubert, *Mathematical problems in image processing: partial differential equations and the calculus of variations*. Springer, 2006, vol. 147.

[29] A. Chambolle, "Partial differential equations and image processing," in *International Conference on Image Processing*, vol. 1. IEEE, 1994, pp. 16–20.

[30] J. B. J. Fourier, *Théorie analytique de la chaleur*. Chez Firmin Didot, père et fils, 1822.

[31] E. M. Stein and R. Shakarchi, *Fourier analysis: an introduction*. Princeton University Press, 2011, vol. 1.

[32] G. B. Folland, *Fourier analysis and its applications*. American Mathematical Soc., 2009, vol. 4.

[33] M. Portnoff, "Time-frequency representation of digital signals and systems based on short-time fourier analysis," *Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 1, pp. 55–69, 1980.

[34] G. Monge, "Mémoire sur la théorie des déblais et des remblais," *Mémoires de Mathématique et de Physique*, pp. 666–704, 1781.

[35] C. Villani, *Optimal transport: old and new*. Springer, 2009, vol. 338.

[36] F. Santambrogio, "Optimal transport for applied mathematicians," *Birkäuser, NY*, vol. 55, no. 58-63, p. 94, 2015.

[37] G. Peyré, M. Cuturi *et al.*, "Computational optimal transport: With applications to data science," *Foundations and Trends® in Machine Learning*, vol. 11, no. 5-6, pp. 355–607, 2019.

[38] G. Toscani and C. Villani, "Probability metrics and uniqueness of the solution to the Boltzmann equation for a Maxwell gas," *Journal of Statistical Physics*, vol. 94, no. 3-4, pp. 619–637, (1999).

[39] J. Carrillo and G. Toscani, "Contractive probability metrics and asymptotic behavior of dissipative kinetic equations," *Prepublicacions del Centre de Recerca Matemàtica*, (2007).

[40] G. Gabetta, G. Toscani, and B. Wennberg, "Metrics for probability distributions and the trend to equilibrium for solutions of the Boltzmann equation," *Journal of Statistical Physics*, vol. 81, no. 5-6, pp. 901–934, (1995).

[41] D. H. Hubel and T. N. Wiesel, "Receptive fields of single neurones in the cat's striate cortex," *The Journal of Physiology*, vol. 148, no. 3, p. 574, 1959.

[42] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[44] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, and V. K. Asari, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv preprint arXiv:1803.01164*, 2018.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[46] A. Shafique, G. Cao, Z. Khan, M. Asad, and M. Aslam, "Deep learning-based change detection in remote sensing images: a review," *Remote Sensing*, vol. 14, no. 4, p. 871, 2022.

[47] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images: From pixel-based to object-based approaches," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 80, pp. 91–106, 2013.

[48] R. J. Vanderbei, *Linear programming*. Springer, 2020.

[49] L. V. Kantarovich, "Mathematical methods in the organization and planning of production," *Publication House of the Leningrad State University.[Translated in Management Science]*, vol. 66, pp. 366–422, 1939.

[50] L. V. Kantorovich, "On one effective method of solving certain classes of extremal problems," in *Doklady Akademii Nauk USSR*, vol. 28, 1940, pp. 212–215.

[51] L. Kantorovitch, "On the translocation of masses," *Management science*, vol. 5, no. 1, pp. 1–4, 1958.

[52] L. V. Kantorovich, "On a problem of monge," *Journal of Mathematical Sciences (NY)*, vol. 133, p. 1383, 2006.

[53] L. Kantorovich and S. Rubinshtein, "On a space of totally additive functions," *Vestnik of the St. Petersburg University: Mathematics*, vol. 13, no. 7, pp. 52–59, 1958.

[54] L. N. Vaserstein, "Markov processes over denumerable products of spaces, describing large systems of automata," *Problemy Peredachi Informatsii*, vol. 5, no. 3, pp. 64–72, 1969.

[55] R. Jordan, D. Kinderlehrer, and F. Otto, "The variational formulation of the fokker–planck equation," *SIAM Journal on Mathematical Analysis*, vol. 29, no. 1, pp. 1–17, 1998.

[56] F. Otto, "The geometry of dissipative evolution equations: the porous medium equation," *Partial Differential Equations*, vol. 26, pp. 101–174, 2001.

[57] L. Guibas, Y. Rubner, and C. Tomasi, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, (2000).

[58] C. L. Mallows, "A note on asymptotic joint normality," *The Annals of Mathematical Statistics*, pp. 508–515, 1972.

[59] H. Tanaka, "An inequality for a functional of probability distributions and its application to kac's one-dimensional model of a maxwellian gas," *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, vol. 27, no. 1, pp. 47–52, 1973.

[60] ——, "Probabilistic treatment of the boltzmann equation," *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, pp. 67–105, 1978.

[61] G. B. Dantzig, *Linear Programming and Extensions*. Princeton University Press, 1963.

[62] ——, "Application of the simplex method to a transportation problem," *Activity Analysis and Production and Allocation*, 1951.

[63] L. Ford Jr and D. Fulkerson, *Flows in Networks*. Princeton University Press, 1962.

[64] B. H. Korte, J. Vygen, B. Korte, and J. Vygen, *Combinatorial optimization*. Springer, 2011, vol. 1.

[65] J. Cuesta-Albertos, C. Matràn, S. Rachev, and L. Ruschendorf, "Mass transportation problems in probability theory," *Mathematical Scientist*, vol. 21, no. 1, p. 34, 1996.

[66] J. Cuesta-Albertos, C. Matràn, and J. Rodriguez-Rodriguez, "Approximation to probabilities through uniform laws on convex sets," *Journal of Theoretical Probability*, vol. 16, no. 2, pp. 363–376, 2003.

[67] E. Del Barrio, J. A. Cuesta-Albertos, C. Matràn, and J. M. Rodriguez-Rodriguez, "Tests of goodness of fit based on the l2-wasserstein distance," *Annals of Statistics*, pp. 1230–1239, 1999.

[68] S. T. Rachev, "The Monge–Kantorovich mass transference problem and its stochastic applications," *Theory of Probability & Its Applications*, vol. 29, no. 4, pp. 647–676, 1985.

[69] L. Rüschendorf, "The Wasserstein distance and approximation theorems," *Probability Theory and Related Fields*, vol. 70, no. 1, pp. 117–129, 1985.

[70] A.-S. Sznitman, "Topics in propagation of chaos," in *Ecole d'ètè de Probabilitès de Saint-Flour XIX—1989*. Springer, 1991, pp. 165–251.

[71] H. Spohn, *Large scale dynamics of interacting particles*. Springer Science & Business Media, 2012.

[72] R. L. Dobrushin, "Prescribing a system of random variables by conditional distributions," *Theory of Probability & Its Applications*, vol. 15, no. 3, pp. 458–486, 1970.

[73] F. Malrieu, "Logarithmic Sobolev inequalities for some nonlinear PDE's," *Stochastic Processes and their Applications*, vol. 95, no. 1, pp. 109–132, 2001.

[74] R. L. Dobrushin, "Perturbation methods of the theory of Gibbsian fields," *Lectures on Probability Theory and Statistics*, pp. 1–66, 1996.

[75] M.-F. Chen, "Trilogy of couplings and general formulas for lower bound of spectral gap," in *Probability towards 2000*. Springer, 1998, pp. 123–136.

[76] Y. Ollivier, "Ricci curvature of Markov chains on metric spaces," *Journal of Functional Analysis*, vol. 256, no. 3, pp. 810–864, 2009.

[77] M. Hairer and J. Mattingly, "Spectral gaps in Wasserstein distances and the 2D Navier–Stokes equation," *The Annals of Probability*, vol. 6, pp. 2050–2091, 2006.

[78] M. Hairer, "Exponential mixing properties of stochastic pdes through asymptotic coupling," *Probability Theory and Related Fields*, vol. 124, no. 3, pp. 345–380, 2002.

[79] J. A. Carrillo, M. DiFrancesco, A. Figalli, T. Laurent, and D. Slepcev, "Global-in-time weak measure solutions and finite-time aggregation for nonlocal interaction equations," *Duke Mathematical Journal*, vol. 156, no. 2, pp. 229–271, 2011.

[80] N. Grunewald, F. Otto, C. Villani, and M. G. Westdickenberg, "A two-scale approach to logarithmic Sobolev inequalities and the hydrodynamic limit," in *Annales de l'IHP Probabilitès et Statistiques*, vol. 45, no. 2, 2009, pp. 302–351.

[81] S. Angenent, S. Haker, A. Tannenbaum, and L. Zhu, "Optimal mass transport for registration and warping," *International Journal of Computer Vision*, vol. 60, no. 3, pp. 225–240, (2004).

[82] J. Feydy, B. Charlier, F.-X. Vialard, and G. Peyré, "Optimal transport for diffeomorphic registration," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2017, pp. 291–299.

[83] N. Kolkin, J. Salavon, and G. Shakhnarovich, "Style transfer by relaxed optimal transport and self-similarity," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 051–10 060.

[84] Y. Mroueh, "Wasserstein style transfer," *arXiv preprint arXiv:1905.12828*, 2019.

[85] O. Pele and M. Werman, "Fast and robust earth mover's distances," in *IEEE International Conference on Computer Vision*. IEEE, (2009), pp. 460–467.

[86] G. Auricchio, F. Bassetti, S. Gualandi, and M. Veneroni, "Computing Wasserstein Barycenters via Linear Programming," in *International Conference on Integration of Constraint Programming, Artificial Intelligence, and Operations Research*. Springer, (2019), pp. 355–363.

[87] M. Cuturi and A. Doucet, "Fast computation of Wasserstein barycenters," in *International Conference on Machine Learning*, (2014), pp. 685–693.

[88] R. Bellazzi, A. Codegoni, S. Gualandi, G. Nicora, and E. Vercesi, "The gene mover's distance: Single-cell similarity via optimal transport," *arXiv preprint arXiv:2102.01218*, 2021.

[89] G.-J. Huizing, G. Peyré, and L. Cantini, "Optimal transport improves cell–cell similarity inference in single-cell omics data," *Bioinformatics*, vol. 38, no. 8, pp. 2169–2177, 2022.

[90] A. Tong, "Graph priors, optimal transport, and deep learning in biomedical discovery," Ph.D. dissertation, Yale University, 2021.

[91] M. Arjovsky, L. Bottou, and S. Chintala, "Wasserstein GAN," *arXiv preprint arXiv:1701.07875*, (2017).

[92] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[93] J. Adler and S. Lunz, "Banach Wasserstein GAN," in *Advances in Neural Information Processing Systems*, (2018), pp. 6754–6763.

[94] C. Frogner, C. Zhang, H. Mobahi, M. Araya, and T. Poggio, "Learning with a Wasserstein loss," in *Advances in Neural Information Processing Systems*, (2015), pp. 2053–2061.

[95] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[96] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced wasserstein discrepancy for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 285–10 295.

[97] N. Papadakis, "Optimal transport for image processing," Ph.D. dissertation, Université de Bordeaux; Habilitation thesis, 2015.

[98] J. Orlin, "A faster strongly polynomial minimum cost flow algorithm," in *Proceedings of the Twentieth annual ACM symposium on Theory of Computing*, 1988, pp. 377–387.

[99] G. Auricchio, F. Bassetti, S. Gualandi, and M. Veneroni, "Computing Kantorovich-Wasserstein distances on d-dimensional histograms using $(d + 1)$-partite graphs," in *Advances in Neural Information Processing Systems*, (2018), pp. 5793–5803.

[100] H. Ling and K. Okada, "An efficient Earth Mover's Distance algorithm for robust histogram comparison," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 840–853, (2007).

[101] G. Auricchio, S. Gualandi, and M. Veneroni, "The maximum nearby flow problem," in *Advances in Optimization and Decision Science for Society, Services and Enterprises*. Springer, 2019, pp. 23–33.

[102] R. Sinkhorn, "A relationship between arbitrary positive matrices and doubly stochastic matrices," *The Annals of Mathematical Statistics*, vol. 35, no. 2, pp. 876–879, 1964.

[103] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[104] L. Ambrosio, N. Gigli, and G. Savaré, *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2005.

[105] L. Ambrosio and N. Gigli, "A user's guide to optimal transport," in *Modelling and Optimisation of Flows on Networks*. Springer, 2013, pp. 1–155.

[106] U. Gianazza, G. Savarè, and G. Toscani, "The wasserstein gradient flow of the fisher information and the quantum drift-diffusion equation," *Archive for Rational Mechanics and Analysis*, vol. 194, no. 1, pp. 133–220, 2009.

[107] S. Arnrich, A. Mielke, M. A. Peletier, G. Savaré, and M. Veneroni, "Passing to the limit in a wasserstein gradient flow: from diffusion to reaction," *Calculus of Variations and Partial Differential Equations*, vol. 44, no. 3, pp. 419–454, 2012.

[108] S. Lisini, D. Matthes, and G. Savaré, "Cahn–hilliard and thin film equations with nonlinear mobility as gradient flows in weighted-wasserstein metrics," *Journal of Differential Equations*, vol. 253, no. 2, pp. 814–850, 2012.

[109] Y. V. Prokhorov, "Convergence of random processes and limit theorems in probability theory," *Theory of Probability & Its Applications*, vol. 1, no. 2, pp. 157–214, 1956.

[110] S. T. Rachev, *Probability metrics and the stability of stochastic models*. Wiley, 1991, vol. 269.

[111] A. Pulvirenti and G. Toscani, "Asymptotic properties of the inelastic Kac model," *Journal of Statistical Physics*, vol. 114, no. 5-6, pp. 1453–1480, (2004).

[112] E. Carlen, E. Gabetta, and G. Toscani, "Propagation of smoothness and the rate of exponential convergence to equilibrium for a spatially homogeneous Maxwellian gas," *Communications in Mathematical Physics*, vol. 199, no. 3, pp. 521–546, (1999).

[113] T. Goudon, S. Junca, and G. Toscani, "Fourier-based distances and Berry-Esseen like inequalities for smooth densities," *Monatshefte für Mathematik*, vol. 135, no. 2, pp. 115–136, (2002).

[114] M. Bisi, J. Carrillo, and G. Toscani, "Decay rates in probability metrics towards homogeneous cooling states for the inelastic Maxwell model," *Journal of Statistical Physics*, vol. 124, no. 2-4, pp. 625–653, (2006).

[115] E. Carlen, M. Carvalho, and E. Gabetta, "Central limit theorem for Maxwellian molecules and truncation of the Wild expansion," *Communications on Pure and Applied Mathematics*, vol. 53, no. 3, pp. 370–397, (2000).

[116] L. Baringhaus and R. Grübel, "On a class of characterization problems for random convex combinations," *Annals of the Institute of Statistical Mathematics*, vol. 49, no. 3, pp. 555–567, (1997).

[117] J. Schrieber, D. Schuhmacher, and C. Gottschlich, "Dotmark–A benchmark for Discrete Optimal Transport," *IEEE Access*, vol. 5, pp. 271–282, 2017.

[118] M. Torregrossa and G. Toscani, "Wealth distribution in presence of debts. A Fokker–Planck description," *arXiv preprint arXiv:1709.09858*, (2017).

[119] K. R. Rao and P. C. Yip, *The transform and data compression handbook*. CRC press, 2018.

[120] K. Rao, D. Kim, and J. Hwang, *Fast Fourier transform-algorithms and applications*. Springer Science & Business Media, (2011).

[121] P. J. Davis, *Circulant matrices*. Wiley, 1979.

[122] S. Lang, *Linear algebra*. Springer Berlin, 1987.

[123] T. Oliphant, "NumPy: A guide to NumPy," USA: Trelgol Publishing, 2006–. [Online]. Available: http://www.numpy.org/

[124] R. Flamary and N. Courty, "POT Python Optimal Transport library," 2017. [Online]. Available: https://github.com/rflamary/POT

[125] S. Gualandi, "Package SpatialKWD," 2021. [Online]. Available: https://github.com/eurostat/Spatial-KWD

[126] F. Ricciato and A. Coluccia, "On the estimation of spatial density from mobile network operator data," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2021.

[127] H. A. Dau, A. Bagnall, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, and E. Keogh, "The ucr time series archive," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 6, pp. 1293–1305, 2019.

[128] J. Pereira and M. Silveira, "Learning representations from healthcare time series data for unsupervised anomaly detection," in *IEEE International Conference on Big Data and Smart Computing*. IEEE, 2019, pp. 1–7.

[129] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2017.

[130] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Lstm-based encoder-decoder for multi-sensor anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[131] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[132] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[133] T. K. Vintsyuk, "Speech discrimination by dynamic programming," *Cybernetics*, vol. 4, no. 1, pp. 52–57, 1968.

[134] Y. Liu, J. Chen, S. Wu, Z. Liu, and H. Chao, "Incremental fuzzy c medoids clustering of time series data using dynamic time warping distance," *Plos one*, vol. 13, no. 5, p. e0197499, 2018.

[135] A. Singh, "Review article digital change detection techniques using remotely-sensed data," *International Journal of Remote Sensing*, vol. 10, no. 6, pp. 989–1003, 1989.

[136] T. Bai, L. Wang, D. Yin, K. Sun, Y. Chen, W. Li, and D. Li, "Deep learning for change detection in remote sensing: a review," *Geo-spatial Information Science*, pp. 1–27, 2022.

[137] H. Chen and Z. Shi, "A spatial-temporal attention-based method and a new dataset for remote sensing image change detection," *Remote Sensing*, vol. 12, no. 10, p. 1662, 2020.

[138] P. P. De Bem, O. A. de Carvalho Junior, R. Fontes Guimarães, and R. A. Trancoso Gomes, "Change detection of deforestation in the brazilian amazon using landsat data and convolutional neural networks," *Remote Sensing*, vol. 12, no. 6, p. 901, 2020.

[139] A. Viña, F. R. Echavarria, and D. C. Rundquist, "Satellite change detection analysis of deforestation rates and patterns along the colombia–ecuador border," *AMBIO: A Journal of the Human Environment*, vol. 33, no. 3, pp. 118–125, 2004.

[140] J. Z. Xu, W. Lu, Z. Li, P. Khaitan, and V. Zaytseva, "Building damage detection in satellite imagery using convolutional neural networks," *arXiv preprint arXiv:1910.06444*, 2019.

[141] C. Zhang, P. Yue, D. Tapete, L. Jiang, B. Shangguan, L. Huang, and G. Liu, "A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 166, pp. 183–200, 2020.

[142] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 1, pp. 574–586, 2018.

[143] A. Varghese, J. Gubbi, A. Ramaswamy, and P. Balamuralidhar, "Changenet: A deep learning architecture for visual change detection," in *European Conference on Computer Vision*, 2018, pp. 129–145.

[144] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–14, 2021.

[145] L. Khelifi and M. Mignotte, "Deep learning for change detection in remote sensing images: Comprehensive review and meta-analysis," *IEEE Access*, vol. 8, pp. 126 385–126 400, 2020.

[146] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional siamese networks for change detection," in *IEEE International Conference on Image Processing*, 2018, pp. 4063–4067.

[147] M. Zhang, G. Xu, K. Chen, M. Yan, and X. Sun, "Triplet-based semantic relation learning for aerial remote sensing image change detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 2, pp. 266–270, 2018.

[148] Y. Liu, C. Pang, Z. Zhan, X. Zhang, and X. Yang, "Building change detection for remote sensing images using a dual-task constrained deep siamese convolutional network model," *IEEE Geoscience and Remote Sensing Letters*, vol. 18, no. 5, pp. 811–815, 2020.

[149] D. Peng, Y. Zhang, and H. Guan, "End-to-end change detection for high resolution satellite images using improved unet++," *Remote Sensing*, vol. 11, no. 11, p. 1382, 2019.

[150] H. Jiang, X. Hu, K. Li, J. Zhang, J. Gong, and M. Zhang, "Pga-siamnet: Pyramid feature-based attention-guided siamese network for remote sensing orthoimagery building change detection," *Remote Sensing*, vol. 12, no. 3, p. 484, 2020.

[151] J. Chen, Z. Yuan, J. Peng, L. Chen, H. Huang, J. Zhu, Y. Liu, and H. Li, "Dasnet: Dual attentive fully convolutional siamese networks for change detection in high-resolution satellite images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 1194–1206, 2020.

[152] W. G. C. Bandara and V. M. Patel, "A transformer-based siamese network for change detection," in *IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 207–210.

[153] S. Chen, K. Yang, and R. Stiefelhagen, "Dr-tanet: dynamic receptive temporal attention network for street scene change detection," in *IEEE Intelligent Vehicles Symposium*, 2021, pp. 502–509.

[154] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 539–546.

[155] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4353–4361.

[156] S. Stent, R. Gherardi, B. Stenger, and R. Cipolla, "Detecting change for multi-view, long-term surface inspection," in *Proceedings of the British Machine Vision Conference*, September 2015, pp. 127.1–127.12.

[157] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[158] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *European Conference on Computer Vision*, 2016, pp. 850–865.

[159] A. Ioannidou, E. Chatzilari, S. Nikolopoulos, and I. Kompatsiaris, "Deep learning advances in computer vision with 3d data: A survey," *ACM Computing Surveys CSUR*, vol. 50, no. 2, pp. 1–38, 2017.

[160] Y. Chu, G. Cao, and H. Hayat, "Change detection of remote sensing image based on deep neural networks," in *International Conference on Artificial Intelligence and Industrial Engineering*, 2016, pp. 262–267.

[161] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "siamese" time delay neural network," *Advances in Neural Information Processing Systems*, vol. 6, pp. 737—-744, 1993.

[162] M. Lebedev, Y. V. Vizilter, O. Vygolov, V. Knyaz, and A. Y. Rubis, "Change detection in remote sensing images using conditional adversarial networks." *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 42, no. 2, pp. 565–571, 2018.

[163] W. Zhao, X. Chen, X. Ge, and J. Chen, "Using adversarial network for multiple change detection in bitemporal remote sensing imagery," *IEEE Geoscience and Remote Sensing Letters*, pp. 1–5, 2020.

[164] X. Peng, R. Zhong, Z. Li, and Q. Li, "Optical remote sensing image change detection based on attention mechanism and image difference," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 9, pp. 7296–7307, 2020.

[165] T. Bao, C. Fu, T. Fang, and H. Huo, "Ppcnet: A combined patch-level and pixel-level end-to-end deep network for high-resolution remote sensing image change detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 10, pp. 1797–1801, 2020.

[166] B. Hou, Q. Liu, H. Wang, and Y. Wang, "From w-net to cdgan: Bitemporal change detection via deep learning techniques," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 3, pp. 1790–1802, 2019.

[167] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu, "Change detection based on deep siamese convolutional network for optical aerial images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1845–1849, 2017.

[168] B. Fang, L. Pan, and R. Kou, "Dual learning-based siamese framework for change detection using bi-temporal vhr optical remote sensing images," *Remote Sensing*, vol. 11, no. 11, p. 1292, 2019.

[169] H. Chen, W. Li, and Z. Shi, "Adversarial instance augmentation for building change detection in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2021.

[170] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015, pp. 1–14.

[171] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 834–848, 2017.

[172] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.

[173] T. Liu, L. Yang, and D. Lunga, "Change detection using deep learning approach with object-based image analysis," *Remote Sensing of Environment*, vol. 256, p. 112308, 2021.

[174] M. Liu, Q. Shi, A. Marinoni, D. He, X. Liu, and L. Zhang, "Super-resolution-based change detection network with stacked attention module for images with different resolutions," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–18, 2021.

[175] B. Wu, C. Xu, X. Dai, A. Wan, P. Zhang, Z. Yan, M. Tomizuka, J. Gonzalez, K. Keutzer, and P. Vajda, "Visual transformers: Token-based image representation and processing for computer vision," *arXiv preprint arXiv:2006.03677*, 2020.

[176] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 6000–6010, 2017.

[177] S. Chen, E. Xie, G. Chongjian, R. Chen, D. Liang, and P. Luo, "Cyclemlp: A mlp-like architecture for dense prediction," in *International Conference on Learning Representations*, 2022.

[178] D. Lian, Z. Yu, X. Sun, and S. Gao, "As-mlp: An axial shifted mlp architecture for vision," in *International Conference on Learning Representations*, 2022.

[179] J. Zhang, K. Yang, C. Ma, S. Reiß, K. Peng, and R. Stiefelhagen, "Bending reality: Distortion-aware transformers for adapting to panoramic semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 917–16 927.

[180] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit *et al.*, "Mlp-mixer: An all-mlp architecture for vision," *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 261–24 272, 2021.

[181] H. Touvron, P. Bojanowski, M. Caron, M. Cord, A. El-Nouby, E. Grave, G. Izacard, A. Joulin, G. Synnaeve, J. Verbeek *et al.*, "Resmlp: Feedforward networks for image classification with data-efficient training," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[182] H. Liu, Z. Dai, D. So, and Q. V. Le, "Pay attention to mlps," *Advances in Neural Information Processing Systems*, vol. 34, pp. 9204–9215, 2021.

[183] T. Yu, X. Li, Y. Cai, M. Sun, and P. Li, "S2-mlp: Spatial-shift mlp architecture for vision," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 297–306.

[184] H. Brezis, *Analisi funzionale: teoria e applicazioni*. Liguori Editore Srl, 1986, vol. 9.

[185] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[186] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, "Stand-alone self-attention in vision models," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[187] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Computational Visual Media*, pp. 1–38, 2022.

[188] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2015, pp. 1026–1034.

[189] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[190] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," *A field guide to dynamical recurrent neural networks.*, pp. 237–244, 2001.

[191] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019, pp. 6105–6114.

[192] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[193] N. Liu, J. Han, and M.-H. Yang, "Picanet: Learning pixel-wise contextual attention for saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3089–3098.

[194] M. Lin, Q. Chen, and S. Yan, "Network in network," in *International Conference on Learning Representations*, 2014.

[195] L. Sifre and S. Mallat, "Rigid-motion scattering for texture classification," *Computer Science*, vol. 3559, pp. 501–515, 2014.

[196] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1251–1258.

[197] W. G. C. Bandara and V. M. Patel, "Revisiting consistency regularization for semi-supervised change detection in remote sensing images," *arXiv preprint arXiv:2204.08454*, 2022.

[198] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[199] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[200] Microsoft, "Neural Network Intelligence," 2021. [Online]. Available: https://github.com/microsoft/nni

[201] I. Loshchilov and F. Hutter, "SGDR: Stochastic Gradient Descent with Warm Restarts," in *International Conference on Learning Representations*, 2019.

[202] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: fast and flexible image augmentations," *Information*, vol. 11, no. 2, p. 125, 2020.

[203] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[204] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.

[205] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020.

[206] K. A. Sankararaman, S. De, Z. Xu, W. R. Huang, and T. Goldstein, "The impact of neural network overparameterization on gradient confusion and stochastic gradient descent," in *International Conference on Machine Learning*, 2020, pp. 8469–8479.